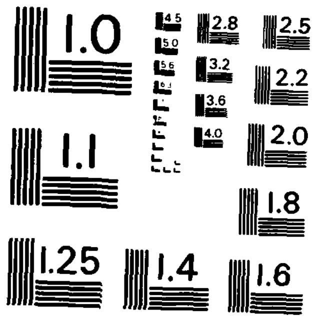
AD-A145 516 AUTO-REGRESSIVE MOVING AVERAGE SPECTRAL ESTIMATION USING MODIFIED AND LEA..(U) NAVAL AIR DEVELOPMENT CENTER WARMINSTER PA SENSORS AND AVIONI.. 1/1

UNCLASSIFIED C D PIZZICHELLO 02 FEB 84 NADC-84027-30 F/G 12/1 NL

END
DATE
FILMED
10-84
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

(12)

REPORT NO. NADC-84027-30

AD-A145 516

AUTO-REGRESSIVE MOVING AVERAGE SPECTRAL ESTIMATION USING MODIFIED AND LEAST SQUARES MODIFIED YULE-WALKER ESTIMATES

Charles D. Pizzichello
Sensors and Avionics Technology Directorate
Naval Air Development Center
Warminster, Pennsylvania 18974

2 FEBRUARY 1984

FINAL REPORT

Approved for Public Release; Distribution Unlimited

Prepared For
NAVAL AIR DEVELOPMENT CENTER
Warminster, Pennsylvania 18974

NOTICES

REPORT NUMBERING SYSTEM – The numbering of technical project reports issued by the Naval Air Development Center is arranged for specific identification purposes. Each number consists of the Center acronym, the calendar year in which the number was assigned, the sequence number of the report within the specific calendar year, and the official 2-digit correspondence code of the Command Office or the Functional Directorate responsible for the report. For example: Report No. NADC-78015-20 indicates the fifteenth Center report for the year 1978, and prepared by the Systems Directorate. The numerical codes are as follows:

CODE	OFFICE OR DIRECTORATE
00	Commander, Naval Air Development Center
01	Technical Director, Naval Air Development Center
02	Comptroller
10	Directorate Command Projects
20	Systems Directorate
30	Sensors & Avionics Technology Directorate
40	Communication & Navigation Technology Directorate
50	Software Computer Directorate
60	Aircraft & Crew Systems Technology Directorate
70	Planning Assessment Resources
80	Engineering Support Group

PRODUCT ENDORSEMENT – The discussion or instructions concerning commercial products herein do not constitute an endorsement by the Government nor do they convey or imply the license or right to use such products.

APPROVED BY: E. Yannuzzi DATE: 1 AUG 1984

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NADC-84027-30	2. GOVT ACCESSION NO. AD-A145 516	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Auto-Regressive Moving Average Spectral Estimation Using Modified and Least Squares Modified Yule-Walker Estimates		5. TYPE OF REPORT & PERIOD COVERED FINAL 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Charles D. Pizzichello		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Sensors & Avionics Technology Directorate, Naval Air Development Center Warminster, PA 18974		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Air Development Center Warminster, PA 18974		12. REPORT DATE 2 February 1984 13. NUMBER OF PAGES 84
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Appendices A, B

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Auto-Regressive Moving Average (ARMA)
Parametric Spectral Estimation
Yule-Walker Equations

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Many popular contemporary spectral estimation methods invoke the modeling of the observation time series by a rational transfer function. These techniques employ the approximation of the second-order statistical relationships using the commonly known Yule-Walker equations. Solving these equations, one obtains the parameter estimates of the hypothesized rational transfer function model. These parameter estimates represent a set of autocorrelation time lags of the observation time series and are ideally selected to optimize the model being considered. Because of the optimization criteria, there are numerous methods/techniques which estimate parameters

20. ABSTRACT (Continued)

that have been proposed in the literature. The objective of this paper is to investigate two of these methods and report their respective performance.

The most general rational transfer function time series model is the Auto-Regressive Moving Average (ARMA) model. This model consists of a rational transfer function containing both poles and zeroes. Of the numerous techniques available in estimating the ARMA parameters, there are two which utilize the Modified Yule-Walker equations. This paper investigates both the Modified Yule-Walker (MYW) and Least Squares Modified Yule-Walker (LSMYW) methods in estimating the ARMA process. The MYW method estimates the ARMA parameters using a minimal set of Yule-Walker equations. In contrast, the LSMYW method utilizes the parametric estimation of an overdetermined set of Yule-Walker equations.

PREFACE

This study was also submitted as a computer simulation project in partial fulfillment for the requirements of a graduate course while working towards the degree Doctor of Philosophy in Electrical Engineering at the University of Rhode Island. This work was done while the author was supported under the Advanced Graduate Study Award Program sponsored by the Naval Air Development Center, Warminster, Pennsylvania.

The author acknowledges the contributions of those individuals who supported the completion of this research. Specifically, he acknowledges the assistance of Professors Donald W. Tufts and Steven M. Kay, both members of the Department of Electrical Engineering at the University of Rhode Island, Kingston, Rhode Island.

The author of this report is located at the
Naval Air Development Center
Warminster, Pennsylvania 18974

ABSTRACT

Many popular contemporary spectral estimation methods invoke the modeling of the observation time series by a rational transfer function. These techniques employ the approximation of the second-order statistical relationships using the commonly known Yule-Walker equations. Solving these equations, one obtains the parameter estimates of the hypothesized rational transfer function model. These parameter estimates represent a set of autocorrelation time lags of the observation time series and are ideally selected to optimize the model being considered. Because of the optimization criteria, there are numerous methods/techniques which estimate parameters that have been proposed in the literature. The objective of this paper is to investigate two of these methods and report their respective performance.

The most general rational transfer function time series model is the Auto-Regressive Moving Average (ARMA) model. This model consists of a rational transfer function containing both poles and zeros. Of the numerous techniques available in estimating the ARMA parameters, there are two which utilize the Modified Yule-Walker equations. This paper investigates both the Modified Yule-Walker (MYW) and Least Squares Modified Yule-Walter (LSMYW) methods in estimating an ARMA process. The MYW method estimates the ARMA parameters using a minimal set of Yule-Walker equations. In contrast, the LSMYW method utilizes the parametric estimation of an overdetermined set of Yule-Walker equations.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

Spectral estimation has received much attention in a variety of applications such as radar, sonar, underwater acoustics, seismology and speech over the last twenty years. The estimation of the power spectral density, or simply the spectrum, is usually based on the use of the Fast Fourier Transform (FFT). Until the introduction of the FFT in 1965, the classical method of estimating the power spectrum was the Blackman and Tukey approach. It should be noted, however, that spectral analysis can be classified into two categories, namely the non-parametric methods and the parametric methods.

The non-parametric methods do not model the spectrum but estimate the power spectrum by performing operations on the observed data set directly. These methods are the classical or traditional techniques in estimating the power spectrum. Examples of these techniques include the Blackman-Tukey, Periodogram, Modified Periodogram and the Averaged Periodogram spectral estimators. The approach of operating on the observed data set directly without modeling is computationally efficient and usually produces reasonable results. However, for certain signal classes, there are inherent performance limitations when a spectrum is estimated by use of traditional methods. Two prominent limitations are frequency resolution and spectral "leakage" caused by windowing the observed data set when the data are processed directly. These two limitations are highlighted when a traditional method is used to analyze short data records. In general, since most data records are short in nature, alternative spectral estimation techniques must be considered.

In contrast to the non-parametric techniques discussed above, the parametric methods assume some particular rational system model that underlies the production of the observation time series. The parametric methods estimate the parameters of an observed data set and then, by substituting these estimates into the theoretical power spectral density implied by the model, yield the computed power spectra. These are the modern methods for estimating power spectra and alternatives to processing data sets directly. Several examples of the models used in implementing these modern methods are: the Auto-Regressive Moving Average (ARMA) model, the Auto-Regressive (AR) model, and the Moving Average (MA) model. Each of these modern parametric estimation methods assume the observed time series is generated from a particular model excited by white gaussian noise.

Of the three models mentioned above, the most general rational system model is the ARMA process. An ARMA model possesses a rational transfer function containing both poles and zeroes. The more specialized models like the MA (all zero) or the AR (all pole) are derived from the general ARMA model. In general, the ARMA model seems to be much more effective than the MA and AR models in modeling an observation time series because it contains both poles and zeroes. However, in practice, most practitioners favor utilizing the specialized MA or AR models, because the ARMA model is computationally more complex, proving to be inefficient in use. This has stimulated considerable activity aimed toward generating computationally efficient ARMA modeling algorithms. Because of this recent activity, several new ARMA modeling algorithms have been proposed and presented in the literature.

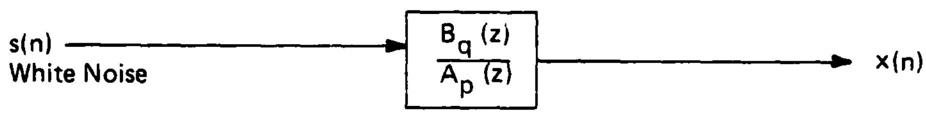
The objective of this paper is to investigate two of these more recent techniques and evaluate their respective performance. These two techniques are the Modified Yule-Walker (MYW) and Least Squares Modified Yule-Walker (LSMYW) approaches to estimating an ARMA model. Like any of the parametric methods discussed above, the ARMA model employs the approximation of second-order statistical relationships which utilize the commonly known Yule-Walker equations. In the case of an ARMA time series, since both poles and zeroes are contained in the rational transfer function model, a special set of Yule-Walker equations should be evaluated. These special equations are

sometimes called the Modified or Extended Yule-Walker equations. To evaluate the MYW and LSMYW approaches in estimating an ARMA model, these Modified Yule-Walker equations will be utilized.

The MYW and LSMYW approaches in estimating an ARMA model and sub-optimal techniques in the sense of practical on-line or real-time processing. These methods estimate an ARMA process by a three-step procedure. This procedure consists of estimating the AR and MA parameters separately rather than jointly as required for the optimal real-time parameter estimation. The procedure first estimates the AR parameters, then inverse filters the original time series by the AR parameters, and finally estimates the MA parameters. This procedure intuitively decouples the AR parameters and the MA parameters, making possible the estimation of each parameter easily but contributes to its sub-optimal nature as a real-time processor. Even though these techniques are sub-optimal, they still make the computational load for estimating ARMA parameters more realizable. It should be noted in this research, the MA parameters will be estimated using Durbin's [1] algorithm.

THEORY

As discussed in the previous section, an ARMA process consists of a rational model transfer function containing poles and zeroes. The order of the numerator designates the number of zeroes q, and the order of the denominator designates the number of poles defined by p. Since an ARMA (p, q) process assumes an ARMA transfer function excited by white gaussian noise, the observation time series x(n) is depicted in Figure 1 as the response or output of a linear time invariant system.



ARMA (p, q)
Transfer Function

Figure 1

The output x(n) can be written as [3]

$$x(n) = -\sum_{k=1}^{p} a(k)\, x(n-k) + \sum_{k=0}^{q} b(k)\, n(n-k) \tag{1}$$

The model's AR parameters (the a(k)'s) and the MA parameters (the b(k)'s) can be obtained directly by a set of exact autocorrelation lags given by

$$R_{xx}(n) = E\,(x(n+k)\; x^*(k)) \tag{2}$$

where E ( ) denotes the expected value operation and x^* denotes the complex conjugate. Since the auto-correlation lags are complex, it can be assumed that the negative lags will be complex conjugate symmetric (i.e., $R_{xx}(-n) = R^*_{xx}(n)$). It should be noted that throughout this paper whenever negative lags are required this property will be automatically assumed.

Therefore, by multiplying both sides of equation (1) by x* (n–L) and taking the expectation yields

$$R_{xx}(L) = -\sum_{k=1}^{p} a(k)R_{xx}(L-k) + \sum_{k=0}^{q} b(k)R_{nx}(L-k) \tag{3}$$

$$\text{where } R_{nx}(L-k) = E\ (n(n)x^*(n-k)) \tag{4}$$

since $R_{nx}(k) = 0$, for $k > 0$, then equation (3) can be written as (3)

$$R_{xx}(L) = -\sum_{k=1}^{p} a(k)R_{xx}(L-k) + \sum_{k=0}^{q} b(k)R_{nx}(L-k) \tag{5a}$$

for L = 0, 1, 2 . . . . . q

$$-\sum_{k=1}^{p} a(k)R_{xx}(L-k) \tag{5b}$$

for L = q+1, q+2 . . . . .

Equations (5) represent the Yule-Walker equations for an ARMA process. Careful examination of the Yule-Walker equations for an ARMA process indicates that the higher order terms do not contain terms of the MA parameters. It seems that for L greater than q, the MA parameters are decoupled from the AR parameters. These higher order equations have been called the Modified or Extended Yule-Walker equations. Expanding the Modified Yule-Walker equations given by equation (5b) yields

$$\begin{vmatrix} R_{xx}(q) & R_{xx}(q-1) & \ldots\ R_{xx}(q-p+1) \\ R_{xx}(q+1) & R_{xx}(q) & \\ \cdot & & \\ \cdot & & \\ \cdot & & \\ R_{xx}(q+p-1) & R_{xx}(q+p-2) & \ldots\ R_{xx}(q) \end{vmatrix} \begin{vmatrix} a_1 \\ a_2 \\ \cdot \\ \cdot \\ \cdot \\ a_p \end{vmatrix} = \begin{vmatrix} -R_{xx}(q+1) \\ -R_{xx}(q+2) \\ \\ \\ \\ -R_{xx}(q+p) \end{vmatrix} \tag{6}$$

in maxtrix notation, they can be written more compactly as

$$R_{xx}' \, a = -R_{xx} \tag{7}$$

The previous section described the two methods being investigated in this paper namely, the MYW and LSMYW techniques. Since both these methods compute the AR parameters first, then compute the MA parameters second, the theory for computing the AR parameters for these two methods is presented.

The MYW method computes the AR parameters by using a "minimal" set of Yule-Walker equations. It utilizes the generation of the Modified Yule-Walker equations given by equation (7). Using this approach, the matrix R'_{xx} of equation (7) will be Toeplitz, square, non-symmetric and not assured to be either positive-definite or non-singular. In order to solve for the AR parameters, a simple matrix inversion of R'_{xx} yields

$$a = -R'^{-1}_{xx} R_{xx} \tag{8}$$

The LSMYW method computes the AR parameters by using an "overdetermined" set of Yule-Walker equations. This method was studied by James A. Cadzow in 1979 [2]. Cadzow's method hypothesizes an overdetermined parametric approach which reduces the undesired hypersensitivity caused by the minimal set of Yule-Walker equations. Using this approach, however, causes the R'_{xx} data matrix from equation (7) to be rectangular or non-square. This requires various least-squares techniques to be implemented which compute the pseudoinverse of the rectangular data matrix R'_{xx} and allow the solving of equation (8) for the AR parameters. The overdetermined input data matrix using the Modified Yule-Walker equations may be defined as

$$\begin{vmatrix} R(q) & R(q-1) & \dots & R(q-p+1) \\ \cdot & & & \\ \cdot & & & \\ R(M-1) & R(M-2) & & R(M-p) \end{vmatrix} \begin{vmatrix} a_1 \\ a_2 \\ \\ a_p \end{vmatrix} = \begin{vmatrix} -R(q+1) \\ -R(q+2) \\ \\ -R(M) \end{vmatrix} \tag{9}$$

where $M >> q+p$

in matrix notation, they can be written more compactly as

$$R'_M a = -R_M \tag{10}$$

To solve equation (10), since R'_M is non-square, a least squares procedure should be utilized. Defining equation (10) in the least squares sense yields

$$R'^H_M R'_M a = -R'^H_M R_M \tag{11}$$

where H is the complex conjugate transpose.

Solving for the AR Parameters using equation (11) gives

$$a = -(R'^H_M R'_M)^{-1} R'^H_M R_M \tag{12}$$

Now that the computation of the AR parameters for both the MYW and LSMYW methods has been defined, our attention will now focus on computing the MA parameters. Recall that in both methods, the next step requires the filtering of the original observation time series of the AR components. This allows the computation of the MA parameters to be calculated separately and efficiently. Therefore, utilizing a finite-impulse-response (FIR) filter and setting its filter coefficients to the AR parameters, the original observation time series can be filtered yielding a pure MA process. This all-zero filter will be defined as

$$Y(n) = \sum_{k=0}^{p} a(k)\, x(n-k) \tag{13}$$

where x(n-k) is the original observation time series a(k) is the AR parameters.

The third and final step is now to compute the MA parameters using some appropriate techniques. In this research, Durbin's algorithm was utilized which computes both the MA parameters and the variance (s^2) associated with that process.

Once the parameters of an ARMA (p,q) process are computed, including the variance (s^2) of the MA process, the power spectral density can be computed by [3]

$$P_x(f) = |H(\exp(j2\pi f)|^2 P_n(f) \tag{14}$$

$$= S^2 \frac{\left| 1 + \sum_{k=1}^{q} b(k)\exp(-j2\pi fk) \right|^2}{\left| 1 + \sum_{k=1}^{p} a(k)\exp(-j2\pi fk) \right|^2}$$

Thus far, all the computations of the ARMA process have been derived knowing the exact auto-correlation lags of the process. However, when a practical implementation is simulated, an estimate of the auto-correlations must be computed. In this research, the unbiased auto-correlation estimate function was used in estimating the auto-correlation lag time series. This estimate auto-correlation function is defined as

$$\hat{R}(k) = \frac{1}{N-k} \sum_{t=0}^{N-1-k} x^*(t)\, x(t+k) \tag{15}$$

The hat (^) over the R in equation (15) denotes an estimate.

SIMULATION PROCEDURE

The investigation of restimating an ARMA process utilizing the MYW and LSMYW methods was performed using specially developed software on an International Business Machines Corporation (IBM) 370 mainframe central computer system. The FFT and matrix inversion computations were accomplished using the International Mathematical and Statistical Library (IMSL). The specialized plotting routines were developed using the COMPLOT software algorithms. Figure 2 depicts the overall software/system flow diagram.

The procedure for evaluating each respective method was to first generate an ARMA model time series given a set of ARMA coefficients. After generating the ARMA time series, the ARMA coefficients were estimated using both techniques, then compared with the original set of given ARMA coefficients. Table 1 depicts the given initial coefficient test conditions for an ARMA (4, 2) process for four unique test cases. A somewhat better presentation and easier interpretation of these initial test conditions can be achieved by calculating the pole/zero locations for each unique test case. The initial pole/zero test conditions were computed and are listed in Table 2. Examining Table 2 shows that Case #1 and Case #2 have moderately strong poles, while Case #3 and Case #4 exhibit very strong poles. The radius is very close to the unit circle. In contrast, with respect to zero locations, Case #2 and Case #3 are strong while Case #1 and Case #4 are weak. An overview of Table 2 yields the features of each unique test case shown in Table 3.

These features play an important role in generating the ARMA (4, 2) time series and also affect performance results as discussed later.

The first step of the simulation procedure consisted of the generation of random White Gaussian Noise (WGN) which was used to generate the ARMA (4, 2) time series. In generating the ARMA (4, 2) times series, the impulse response of an infinite pulse response (IIR) filter with the ARMA (4, 2) coefficients was computed. Then the impulse response and the WGN were convolved to generate the ARMA (4, 2) time series. It should be noted when implementing an IIR filter, in theory its length is infinite; however, in order to implement an IIR filter in practice, only a finite filter length should first be computed to some finite decimal point accuracy. The finite length of the IIR filter computed in this research was computed by

$$r^n = 0.00002 \tag{16}$$

where r = radius of pole

n = finite filter length

The finite length is very dependent on the closeness of the pole to the unit circle. Therefore, in computing the finite filter length, the strongest pole radius for each case was used. For Cases #1 and #2, n was equal to 40, while for Cases #3 and #4, n was set to 500.

Another important point of interest is when the convolution process occurs, the impulse response sequence is flipped about the origin and then slides and multiplies the WGN. This flipping operation causes the index of the finite filter response to go from positive to negative.

This causes the initial values of the convolution process to be non-stationary and contain transient spikes. To generate a stationary ARMA (4, 2) time series, these initial time series values should be filtered out. The number of points filtered out or thrown away should be greater than the finite filter length of the IIR filter response. For added safety, assuming worst case conditions for all test cases, the first 1000 time series values were discarded to ensure stationarity in all simulation runs.

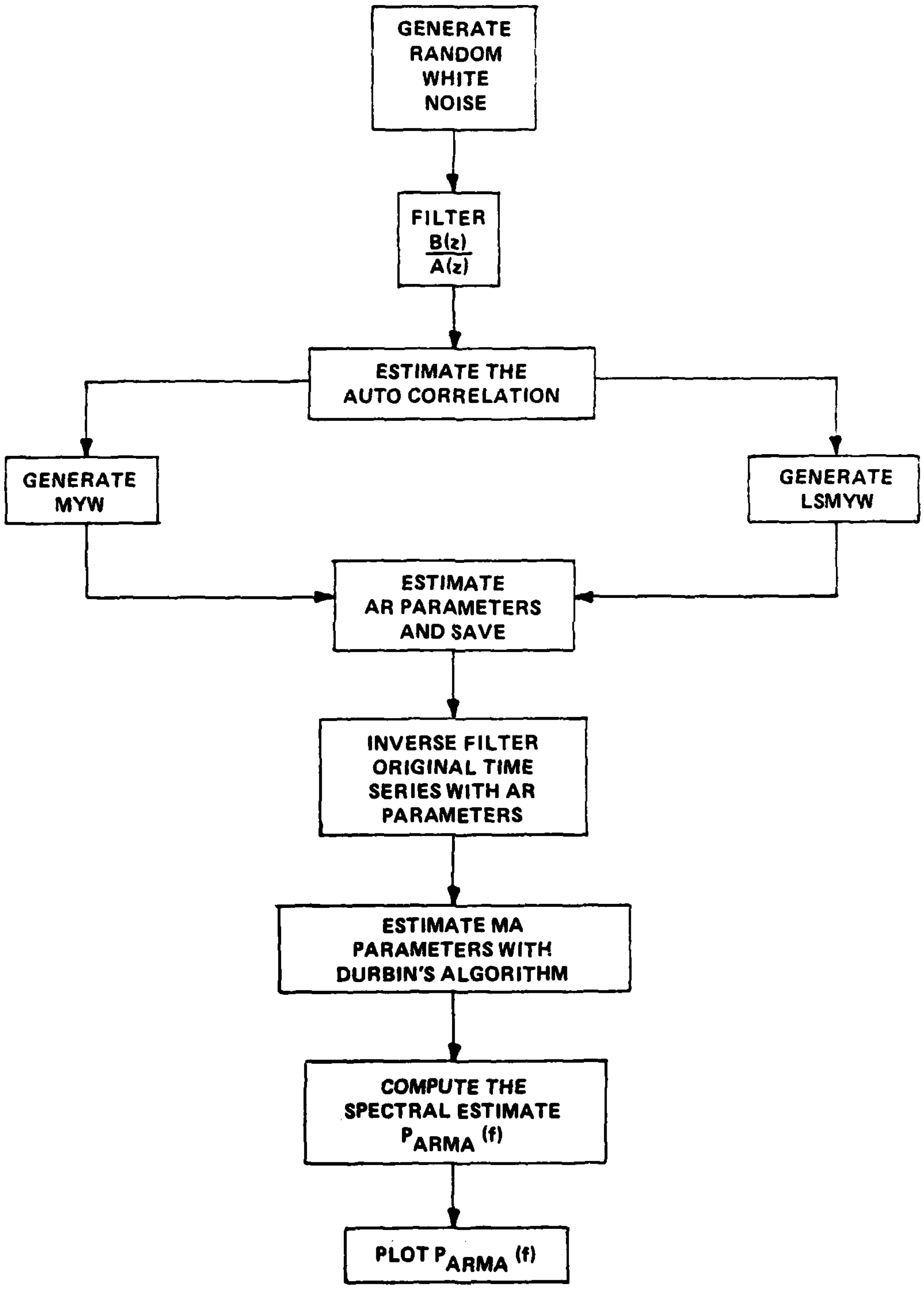


Figure 2. Software Simulation Flow Diagram

Table 1. Initial Coefficient Test Conditions

	A1	A2	A3	A4	B1	B2
CASE #1	-1.352	1.338	-.662	.240	-.2	.04
CASE #2	-1.352	1.338	-.662	.240	-.9	.81
CASE #3	-2.7604	3.8085633	-2.6535	.9238	-.9	.81
CASE #4	-2.7604	3.8085633	-2.6535	.9238	-.2	.04

Table 2. Initial Pole/Zero Test Conditions

	POLE #1		POLE #2		ZERO #1	
	RADIUS	FREQUENCY	RADIUS	FREQUENCY	RADIUS	FREQUENCY
CASE #1	.700	0.125	.699	0.208	.200	0.166
CASE #2	.700	0.125	.699	0.208	.900	0.166
CASE #3	.979	0.109	.981	0.140	.900	0.166
CASE #4	.979	0.109	.981	0.140	.200	0.166

NOTE: ALL POLE/ZERO VALUES LISTED ABOVE ALSO HAVE SYMMETRIC COMPLEX CONJUGATE VALUES

Table 3

	POLES	ZERO
Case #1	Moderately Strong	Weak
Case #2	Moderately Strong	Strong
Case #3	Strong	Strong
Case #4	Strong	Weak

After the ARMA time series had been generated, the next step was to estimate the auto-correlation of the generated time series. In this research, the unbiased auto-correlation estimate function was computed using equation (15) from the previous section. Following the estimation of the auto-correlation function, both the MYW and LSMYW equations could now be generated from the correlation estimates. In the case of the MYW method, the minimum number of correlation time series lags required to form these equations is p+q (the order of both the AR and MA parameters). During simulation runs, 25 time series lags were computed. When the LSMYW equations are computed, an overdetermined set of correlation time series lags are required. This length must be much greater than p+q and is designated M1. During LSMYW simulation runs, M1 was set at 50, and 75 time series lags were computed. For all simulation runs, the length of the input data time series was 256 points.

The next step was to estimate the AR parameters from the newly formed MYW and LSMYW equations. Both methods require a matrix inversion which provides the AR parameter estimates.

In the LSMYW approach, an intermediate step had to be performed prior to computing its matrix inversion. This additional step was required because in the LSMYW approach, the correlation matrix is non-square, therefore requiring a pseudoinverse of the correlation data matrix which allowed solving the AR parameters.

Next the original time series was filtered by a FIR filter using the AR parameters as filter coefficients. This filtering operation yielded a time series containing only MA parameters. Then the MA parameters could be estimated using Durbin's algorithm which includes the estimates of the MA parameters and the variance associated with that process. The final step was to compute the power spectral density using the estimated AR parameters, MA parameters and variance and plot the results.

RESULTS AND CONCLUSIONS

The results of the simulations of the MYW and LSMYW methods in estimating the parameters of an ARMA (4, 2) process are contained in Appendix A. For each test case of each method, there were three experiments which yielded three unique plot presentations. These three presentations are (1) the true power spectrum vs the estimated spectrum for one realization, (2) the true spectrum vs an average of 50 estimated realizations and (3) the multiple plot of 50 estimated events. In addition, there is also a plot of the true spectrum, a true pole/zero plot and the estimated pole/zero plot for one realization for each test case. For all 50 event runs, there was no redundancy or overlap of the input time series values from one time-frame to the next.

Table 4 shows the estimated coefficients for one realization of the LSMYW method, while Table 5 lists the estimated coefficients of one realization of the MYW method. Comparing the results of Table 4 and 5 with Table 1, indicates that in general the LSMYW method is superior to the MYW method in estimating the parameters of an ARMA process.

The LSMYW method may be assumed to be superior to the MYW method because the LSMYW method uses an overdetermined set of correlation lags. For the cases where the poles are close to the unit circle, the LSMYW method should outperform the MYW method.

A careful review of all the results contained in Appendix A reveals that both methods exhibit hypersensitivity caused by the closeness of either the pole or zero radius to the unit circle. Cases #3 and #4, the strong pole location test cases, seem to exhibit the best results for both methods. However, the LSMYW method outperforms the MYW method as expected. In Cases #1 and #2, both

Table 4. One Realization of Least Squares Modified Yule-Walker Estimates

	A1	A2	A3	A4	B1	B2
CASE #1	-1.8444	2.1362	-1.2857	0.5029	-0.6799	0.2576
CASE #2	-1.2690	1.0800	-0.6429	0.3318	-0.7268	0.3946
CASE #3	-2.7511	3.8050	-2.6473	0.9241	-0.8457	0.6776
CASE #4	-2.7122	3.7443	-2.6036	0.9207	-0.1346	0.0663

Table 5. One Realization of Modified Yule-Walker Estimates

	A1	A2	A3	A4	B1	B2
CASE #1	-0.1435	1.4551	-0.8115	0.4746	0.0340	0.4338
CASE #2	-1.2672	1.4643	-0.8598	0.2371	-0.6344	0.8144
CASE #3	-2.2047	3.0046	-2.1169	0.9251	-0.6242	0.8997
CASE #4	-2.7097	3.5733	-2.3706	0.7619	-0.2122	-0.1136

methods suffer decreased performance which is attributed to the weaker pole locations. The worst case is depicted in Case #2 for either method as seen in the multiple 50 realization plot. This case is the moderately strong pole/strong zero test case which exhibits poor variance estimates using Durbin's algorithm. This poor variance estimate is attributed to the strong zero/moderately strong pole location combination, since the variance estimates for the other test cases were closer to the original variance used to generate the original ARMA (4, 2) time series.

In conclusion, the LSMYW and MYW spectral estimators have been shown to produce reliable ARMA spectral estimates for certain signal classes. In addition, both methods contribute to reducing the computation burden of computing an ARMA spectral estimate making possible its computation more realizable. It has also been shown that the LSMYW method outperforms the MYW method but still suffers the problem of hypersensitivity caused by the close proximity of either the pole or zero location to the unit circle. Therefore, it is recommended that future research address the technical issue of hypersensitivity and also investigate another technique in estimating the processes' variance independent of Durbin's algorithm.

REFERENCES

[1] J. Durbin, "Efficient Estimation of Parameters in Moving Average Models," Biometrika, Vol. 46, p. 306, 1959.

[2] James A. Cadzow, "Spectral Estimation: An Overdetermined Rational Model Equation Approach," IEEE Proceedings, Vol. 70, No. 9, pp. 907-939, September 1982.

[3] S. M. Kay and S. L. Marple, Jr., "Spectrum Analysis – A Modern Perspective," IEEE Proceedings, Vol 69, No. 11, pp. 1380-1419, November 1981.

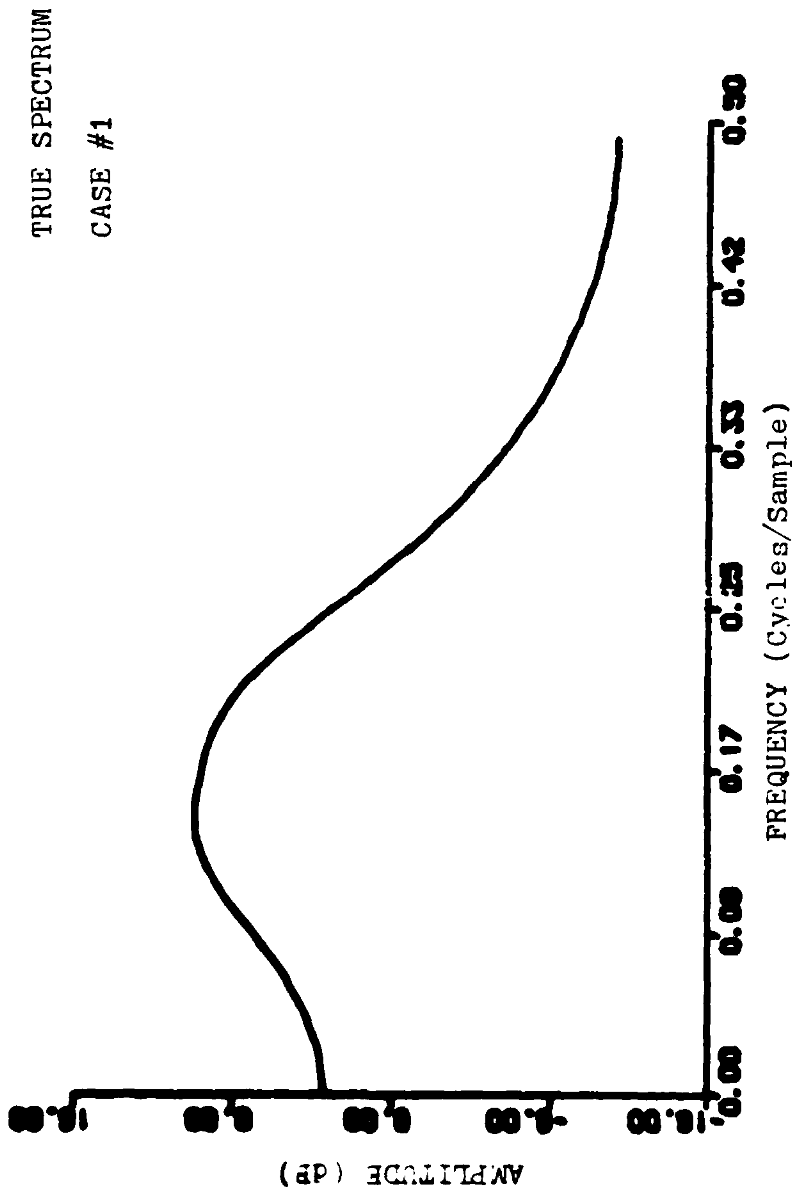
APPENDIX A

PLOTTED EXPERIMENTAL RESULTS

LEGEND FOR APPENDIX A

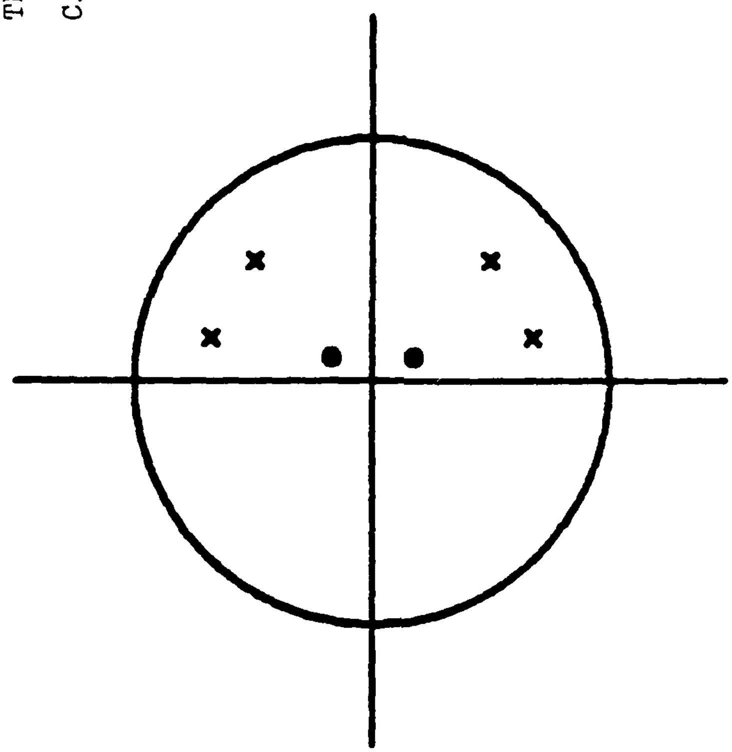
SYMBOL	DESCRIPTION
MYWE	Modified Yule-Walker Equations
LSMYWE	Least Squares Modified Yule-Walker Equations
dB	decibels
X	poles
0	zeroes

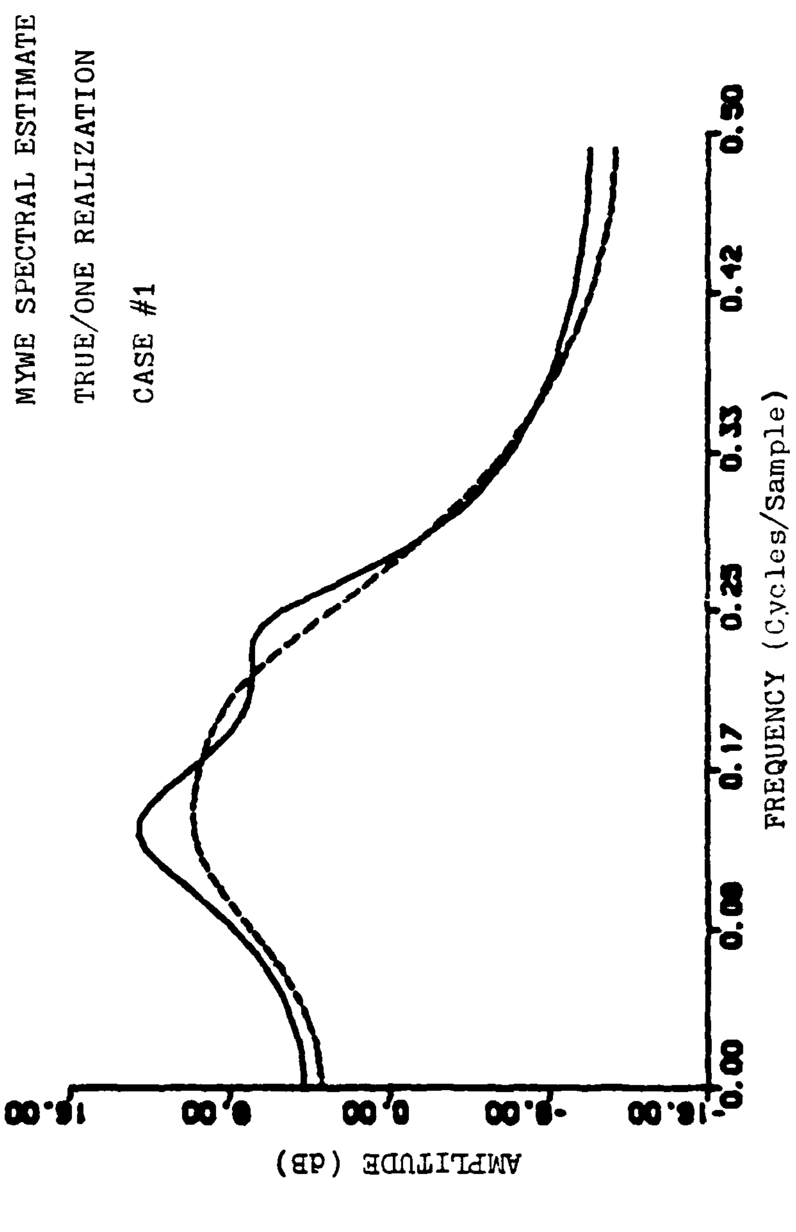
NOTE: ON ALL GRAPHS THAT CONTAIN BOTH CONTINUOUS AND DASHED LINES, THE DASHED LINE REPRESENTS THE TRUE SPECTRUM


TRUE SPECTRUM
CASE #1
FREQUENCY (Cycles/Sample)
AMPLITUDE (dB)
0.00
0.08
0.17
0.25
0.33
0.42
0.50
-16.00
-8.00
0.00
8.00
16.00

TRUE POLE/ZERO PLOT

CASE #1



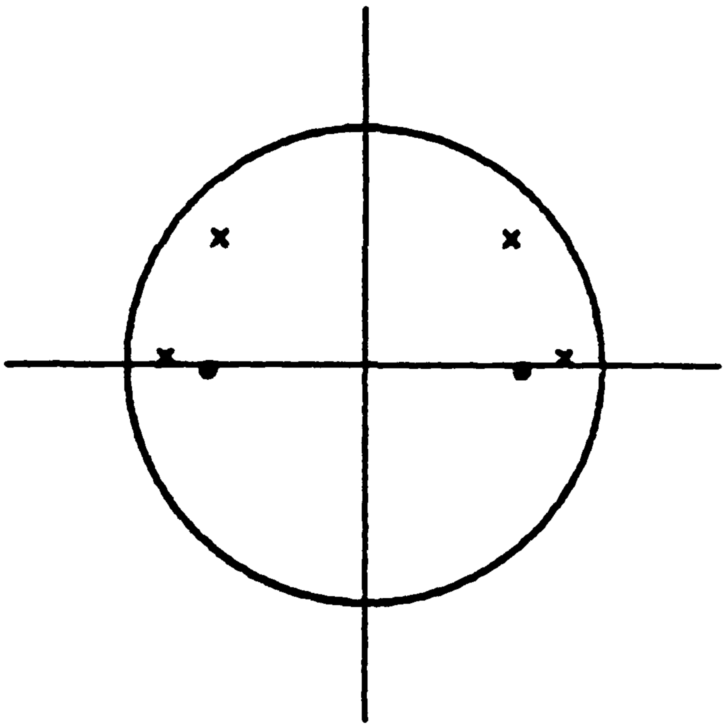

MYWE SPECTRAL ESTIMATE
TRUE/ONE REALIZATION
CASE #1
FREQUENCY (Cycles/Sample)
0.00
0.08
0.17
0.25
0.33
0.42
0.50
AMPLITUDE (dB)
-16.00
-8.00
0.00
8.00
16.00

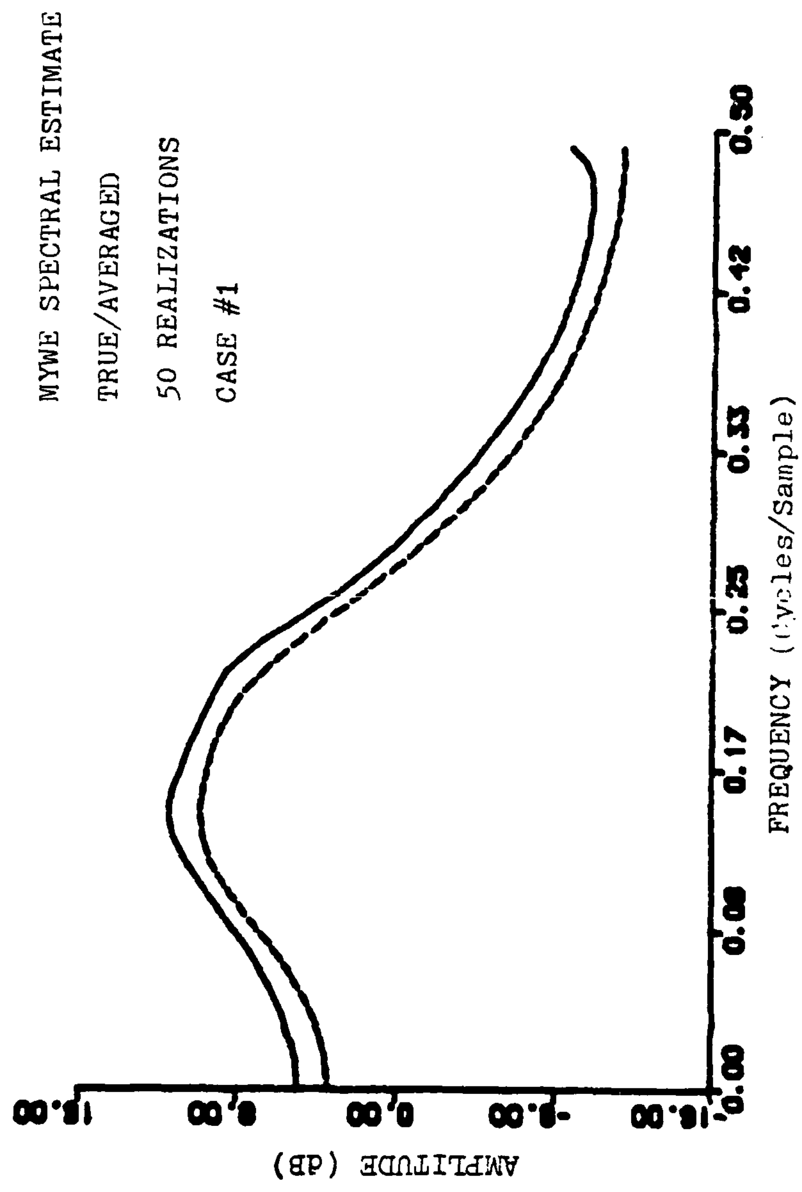
MYWE SPECTRAL ESTIMATE

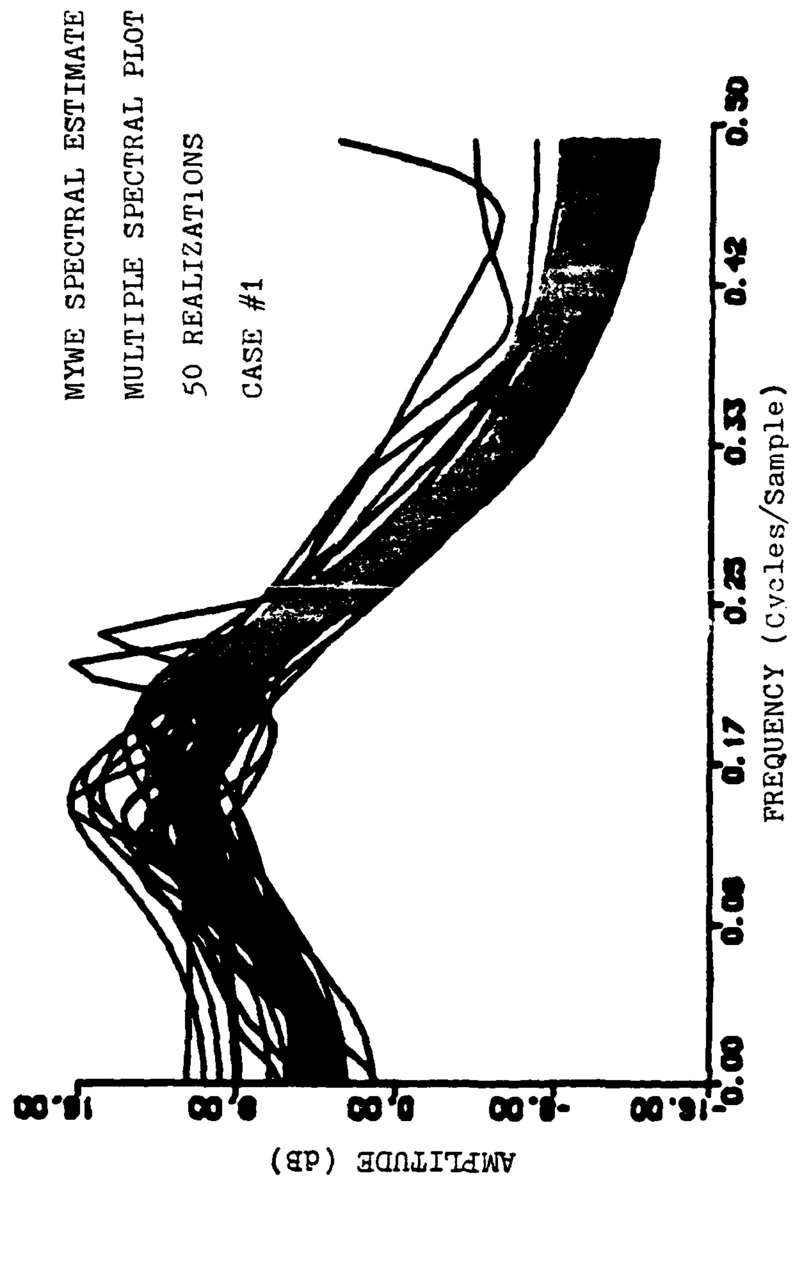
POLE/ZERO PLOT

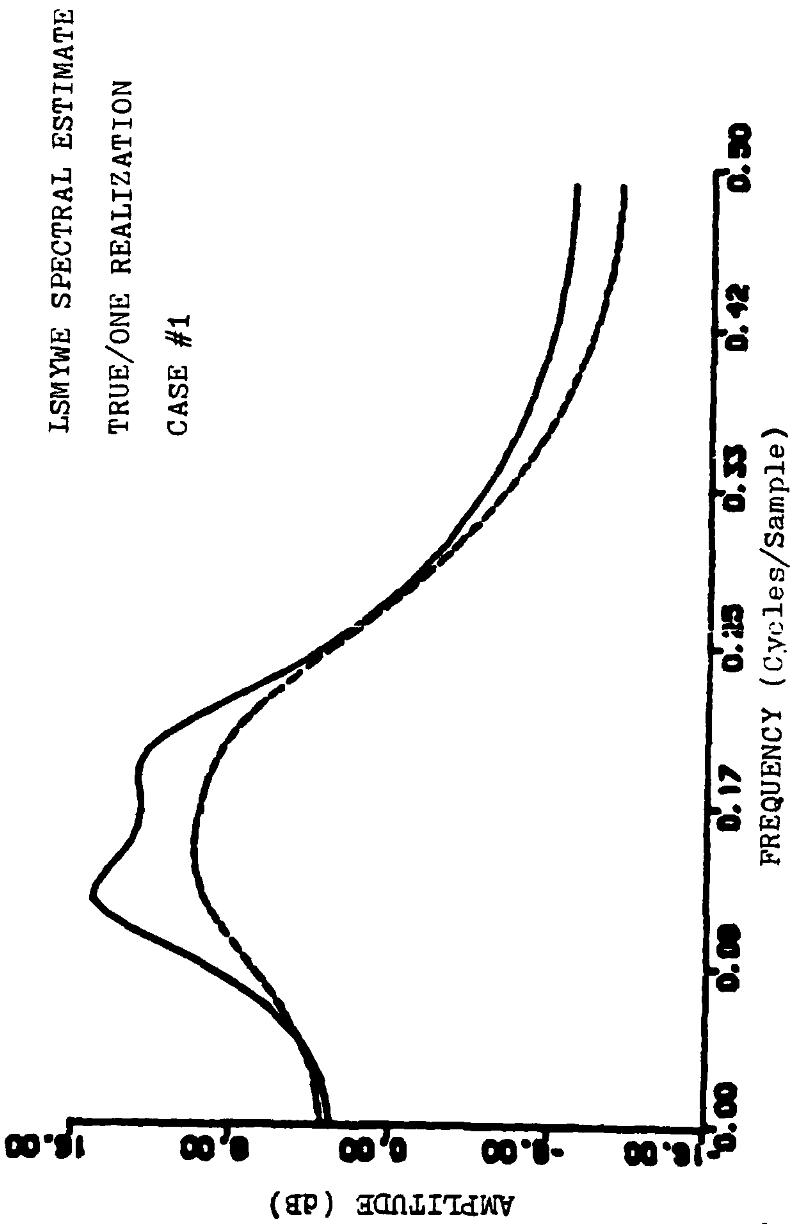
ONE REALIZATION

CASE #1




MYWE SPECTRAL ESTIMATE
TRUE/AVERAGED
50 REALIZATIONS
CASE #1
AMPLITUDE (dB)
16.00
8.00
0.00
-8.00
-16.00
0.00
0.08
0.17
0.25
0.33
0.42
0.50
FREQUENCY (Cycles/Sample)


MYWE SPECTRAL ESTIMATE
MULTIPLE SPECTRAL PLOT
50 REALIZATIONS
CASE #1
AMPLITUDE (dB)
16.00
8.00
0.00
-8.00
-16.00
0.00
0.08
0.17
0.25
0.33
0.42
0.50
FREQUENCY (Cycles/Sample)

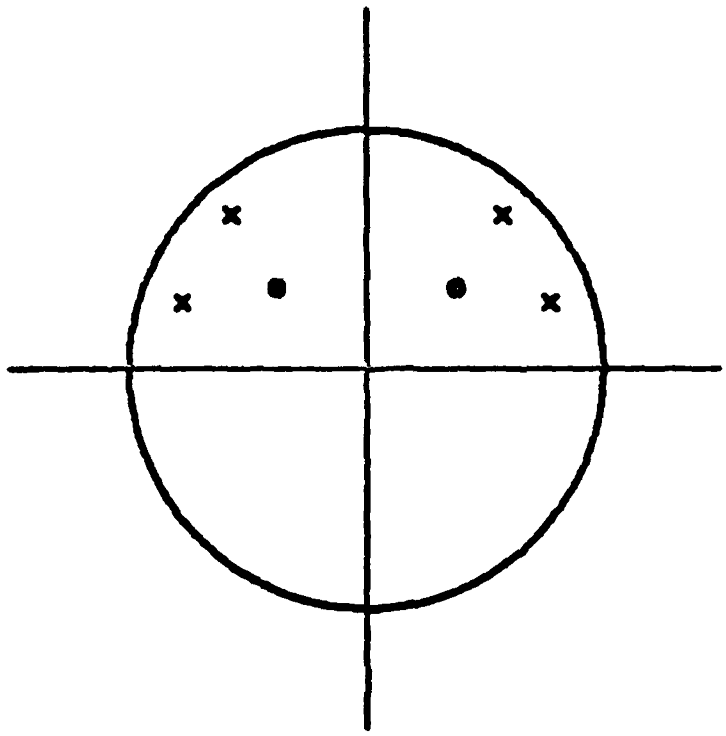

LSMYWE SPECTRAL ESTIMATE
TRUE/ONE REALIZATION
CASE #1
FREQUENCY (Cycles/Sample)
AMPLITUDE (dB)
0.00
0.17
0.42
0.50

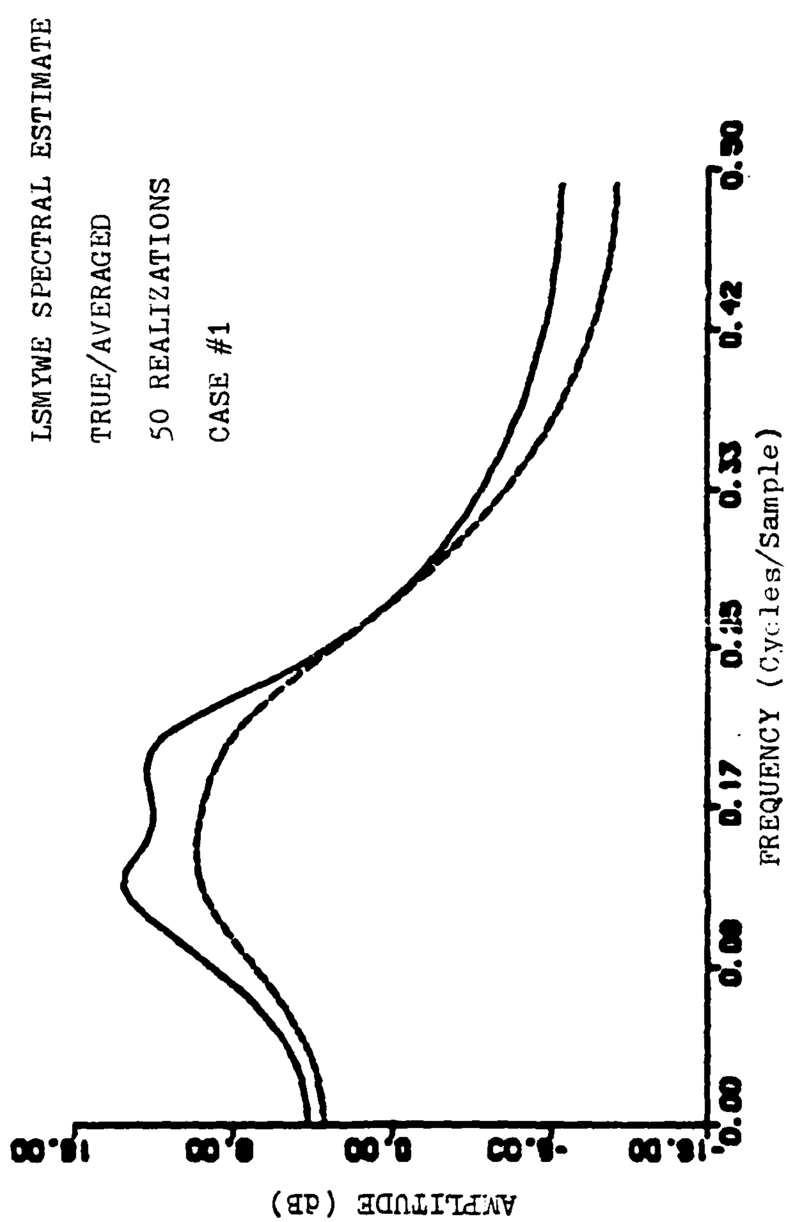
LSMYWE SPECTRAL ESTIMATE

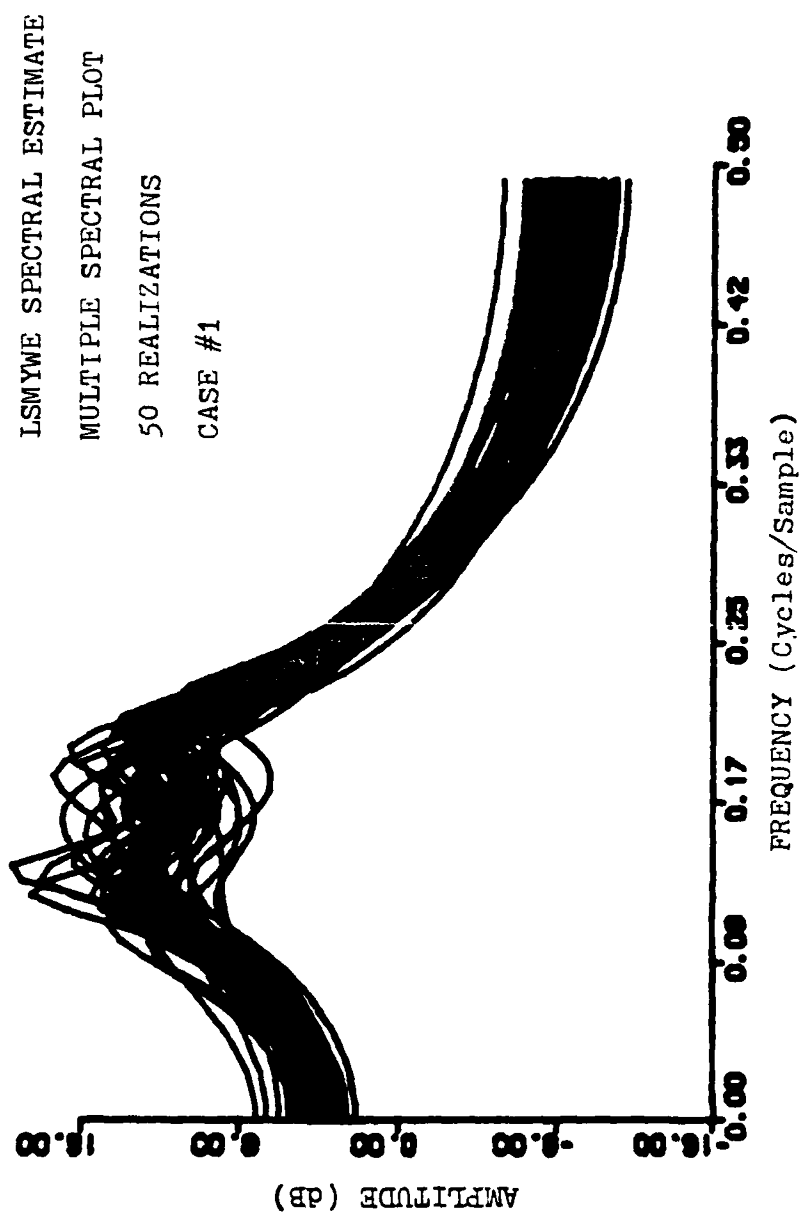
POLE/ZERO PLOT

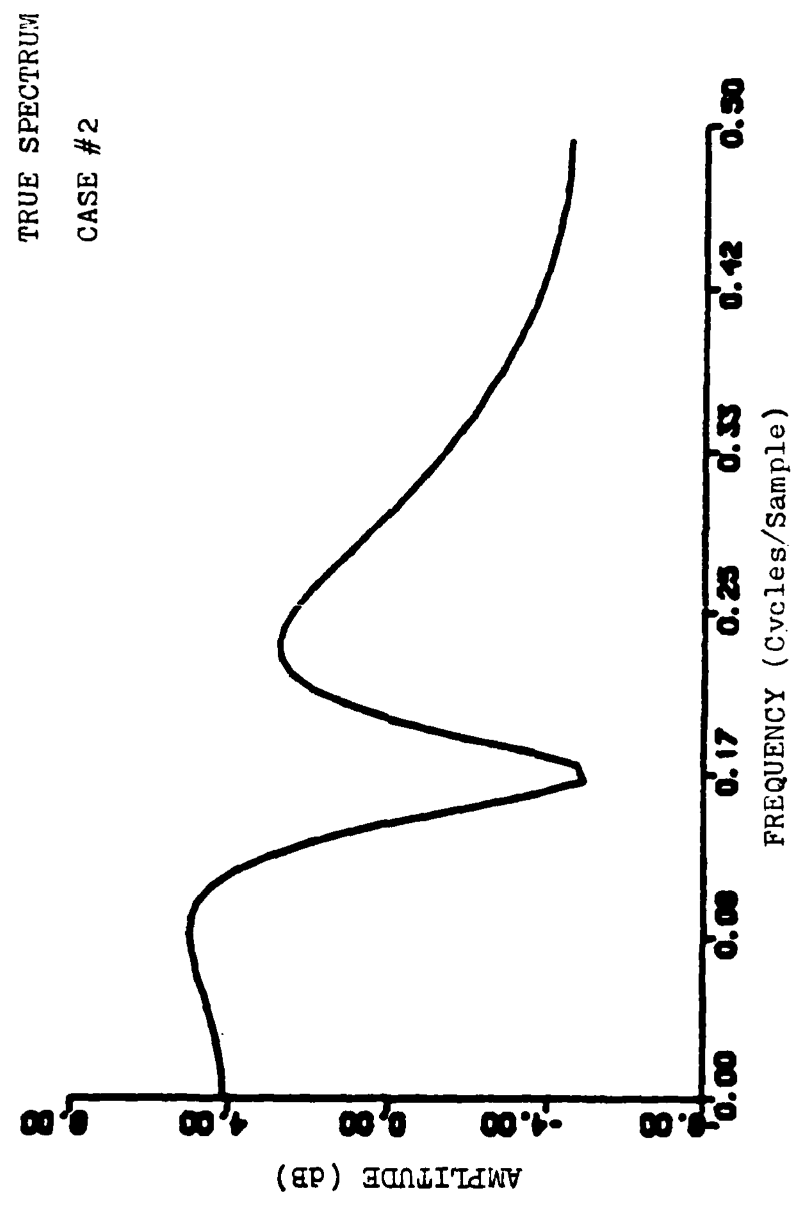
ONE REALIZATION

CASE #1



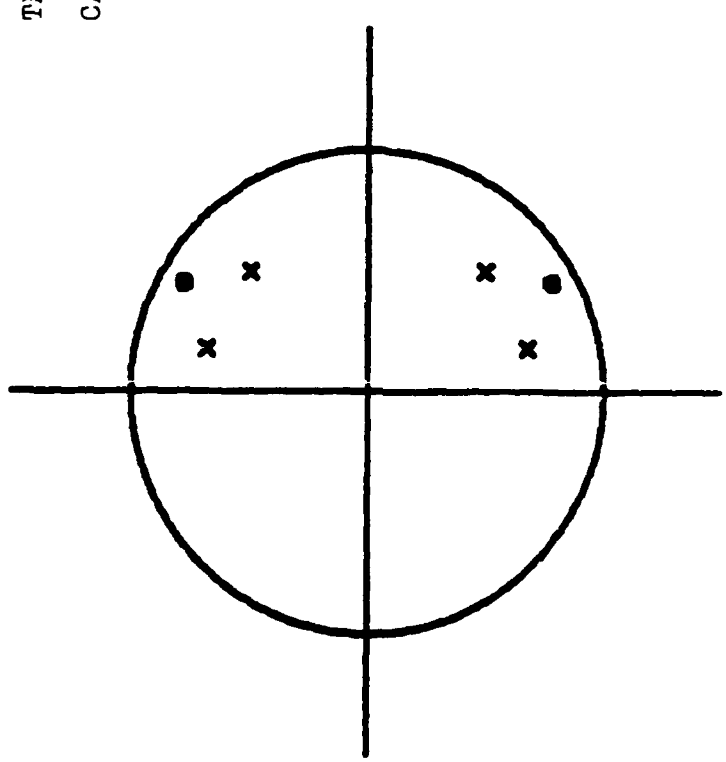

LSMYWE SPECTRAL ESTIMATE
TRUE/AVERAGED
50 REALIZATIONS
CASE #1
16.00
8.00
0.00
-8.00
-16.00
AMPLITUDE (dB)
0.00
0.08
0.17
0.25
0.33
0.42
0.50
FREQUENCY (Cycles/Sample)

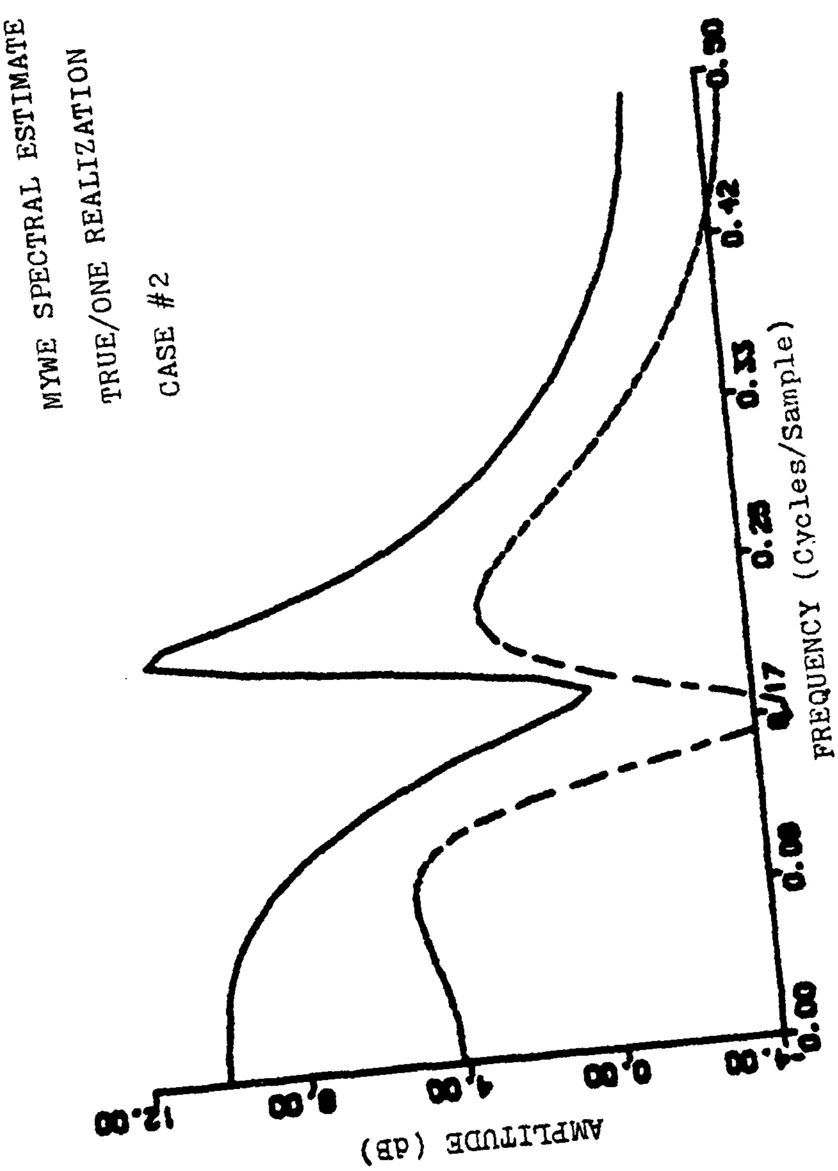

LSMYWE SPECTRAL ESTIMATE
MULTIPLE SPECTRAL PLOT
50 REALIZATIONS
CASE #1
AMPLITUDE (dB)
16.00
8.00
0.00
-8.00
-16.00
0.00
0.08
0.17
0.25
0.33
0.42
0.50
FREQUENCY (Cycles/Sample)


TRUE SPECTRUM
CASE #2
FREQUENCY (Cycles/Sample)
0.00
0.08
0.17
0.25
0.33
0.42
0.50
AMPLITUDE (dB)
8.00
4.00
0.00
-4.00
-8.00

TRUE POLE/ZERO PLOT

CASE #2



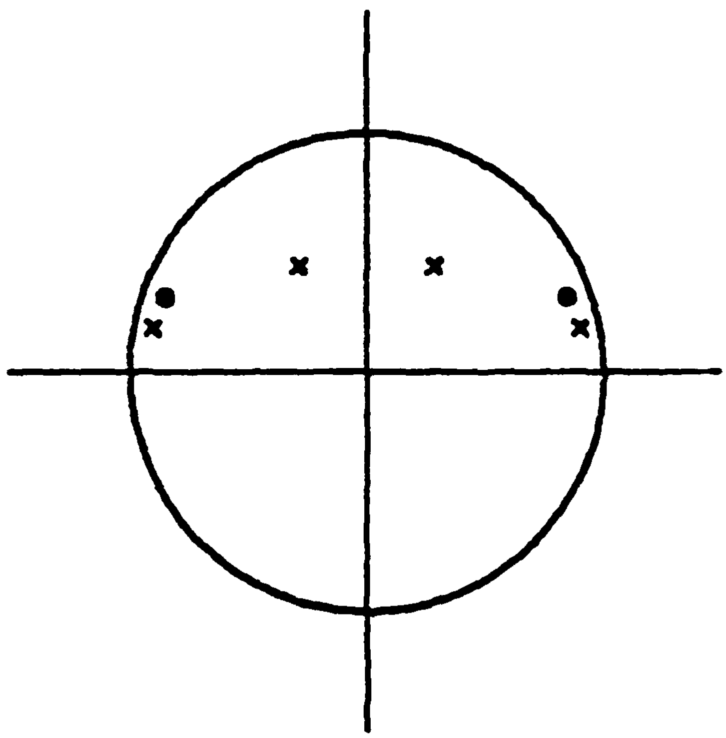

MYWE SPECTRAL ESTIMATE
TRUE/ONE REALIZATION
CASE #2
FREQUENCY (Cycles/Sample)
0.00
0.08
0.17
0.25
0.33
0.42
0.50
AMPLITUDE (dB)
12.00
8.00
4.00
0.00

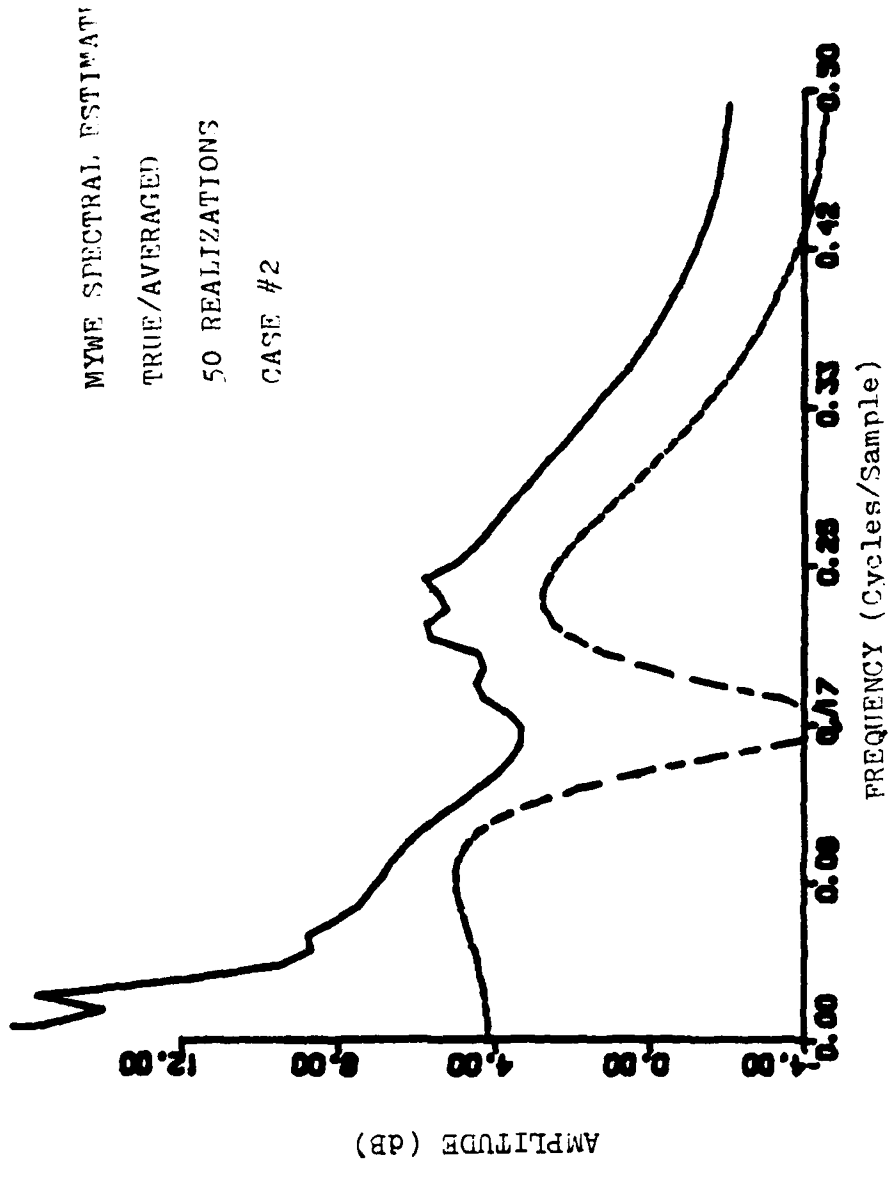
MYWE SPECTRAL ESTIMATE

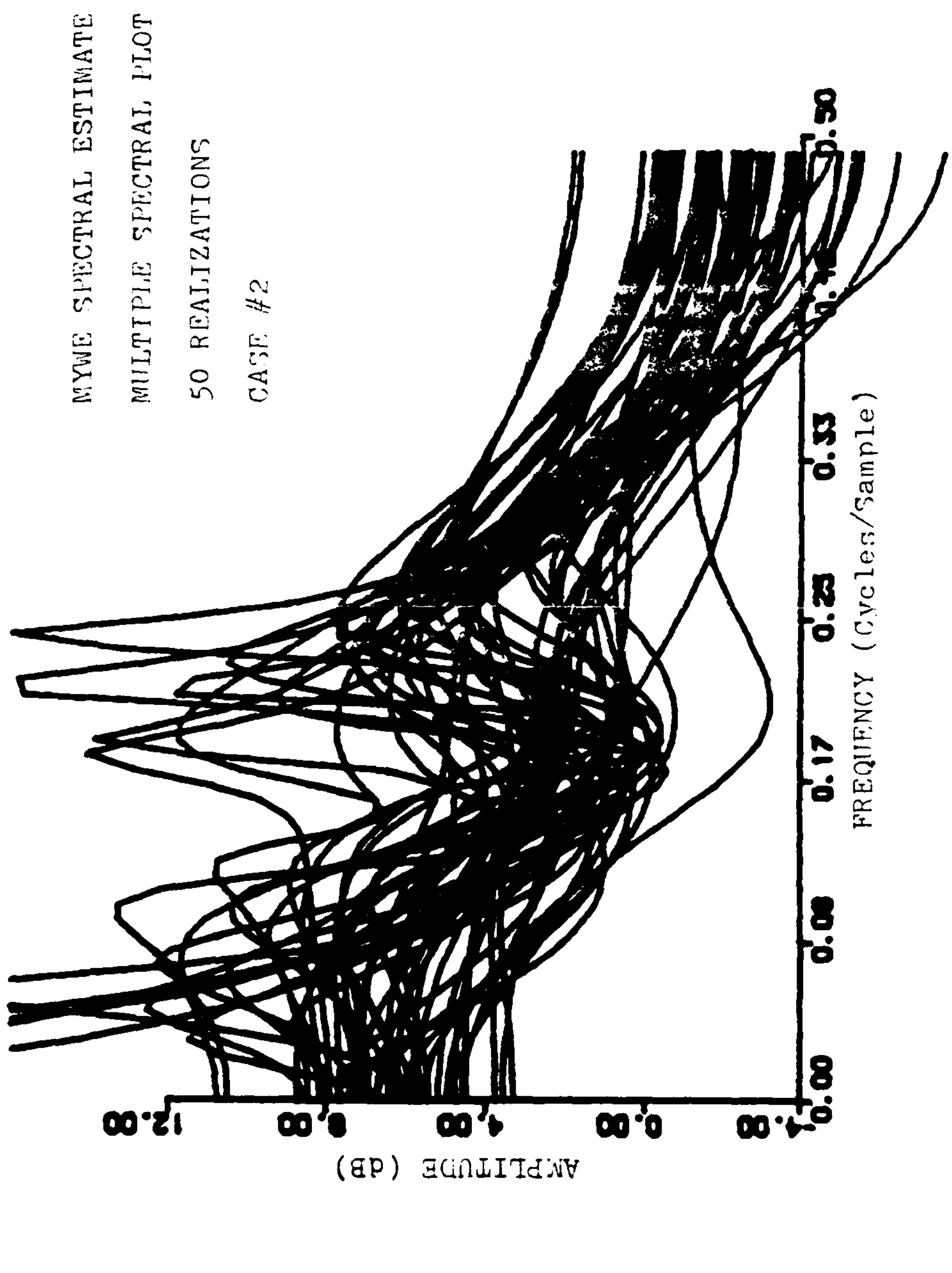
POLE/ZERO PLOT

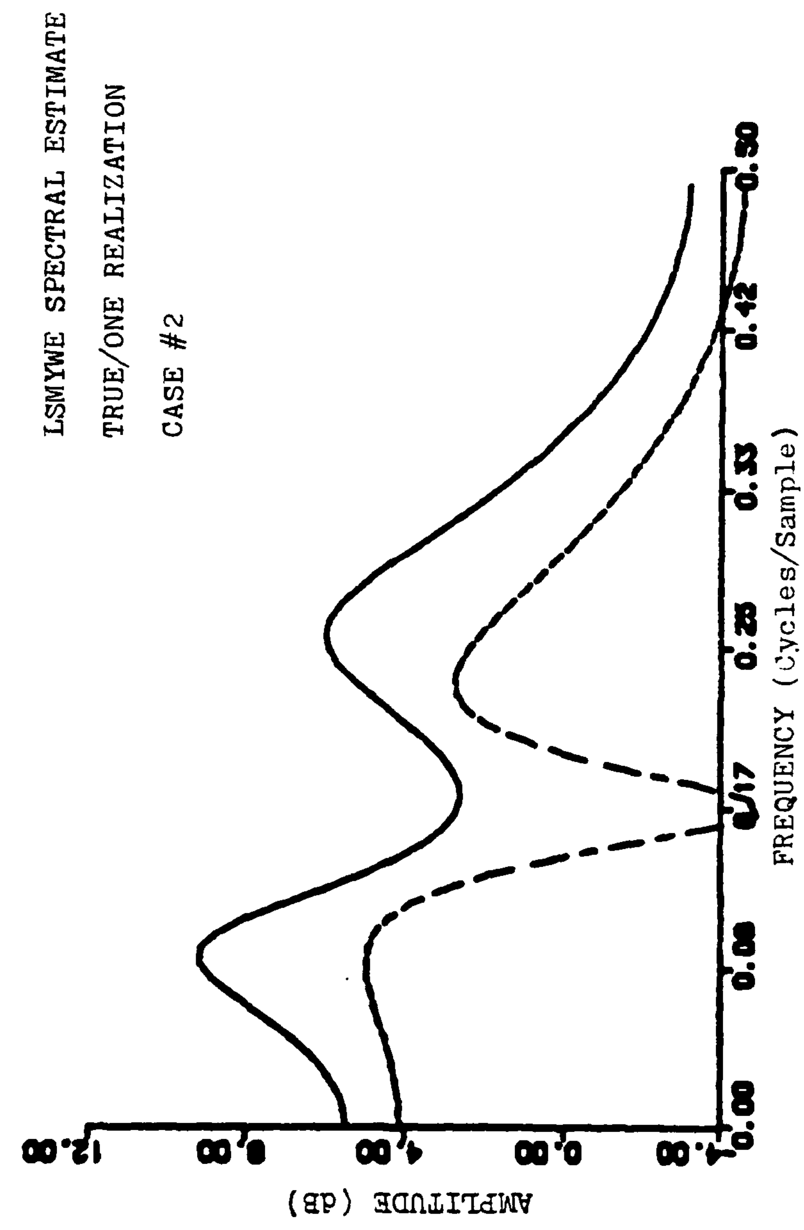
ONE REALIZATION

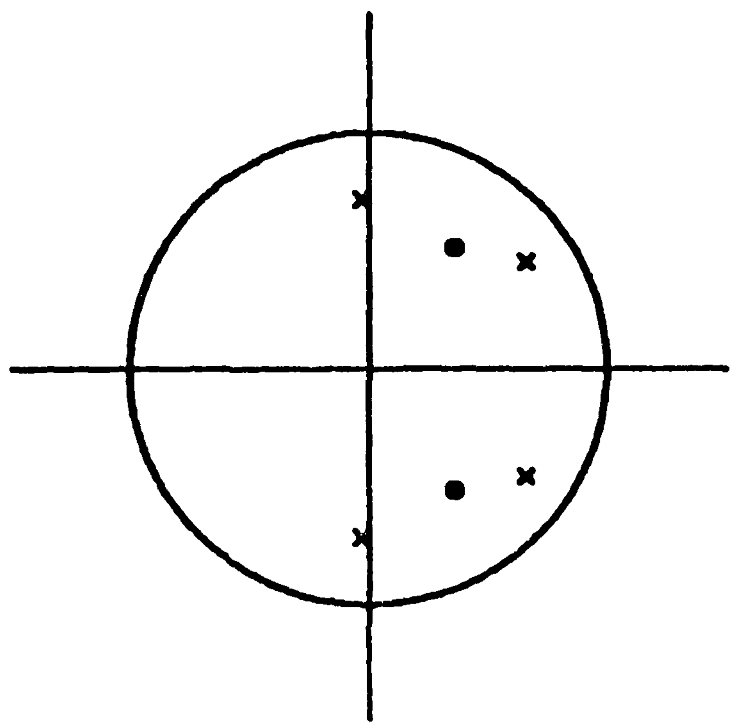
CASE #2




MYWE SPECTRAL ESTIMAT
TRUE/AVERAGED
50 REALIZATIONS
CASE #2
12.00
8.00
4.00
0.00
-4.00
AMPLITUDE (dB)
0.00
0.08
0.17
0.25
0.33
0.42
0.50
FREQUENCY (Cycles/Sample)


MYWE SPECTRAL ESTIMATE
MULTIPLE SPECTRAL PLOT
50 REALIZATIONS
CASE #2
AMPLITUDE (dB)
12.00
8.00
4.00
0.00
-4.00
FREQUENCY (Cycles/Sample)
0.00
0.08
0.17
0.25
0.33
0.50


LSMYWE SPECTRAL ESTIMATE
TRUE/ONE REALIZATION
CASE #2
AMPLITUDE (dB)
12.00
8.00
4.00
0.00
-4.00
0.00
0.08
0.17
0.25
0.33
0.42
0.50
FREQUENCY (Cycles/Sample)

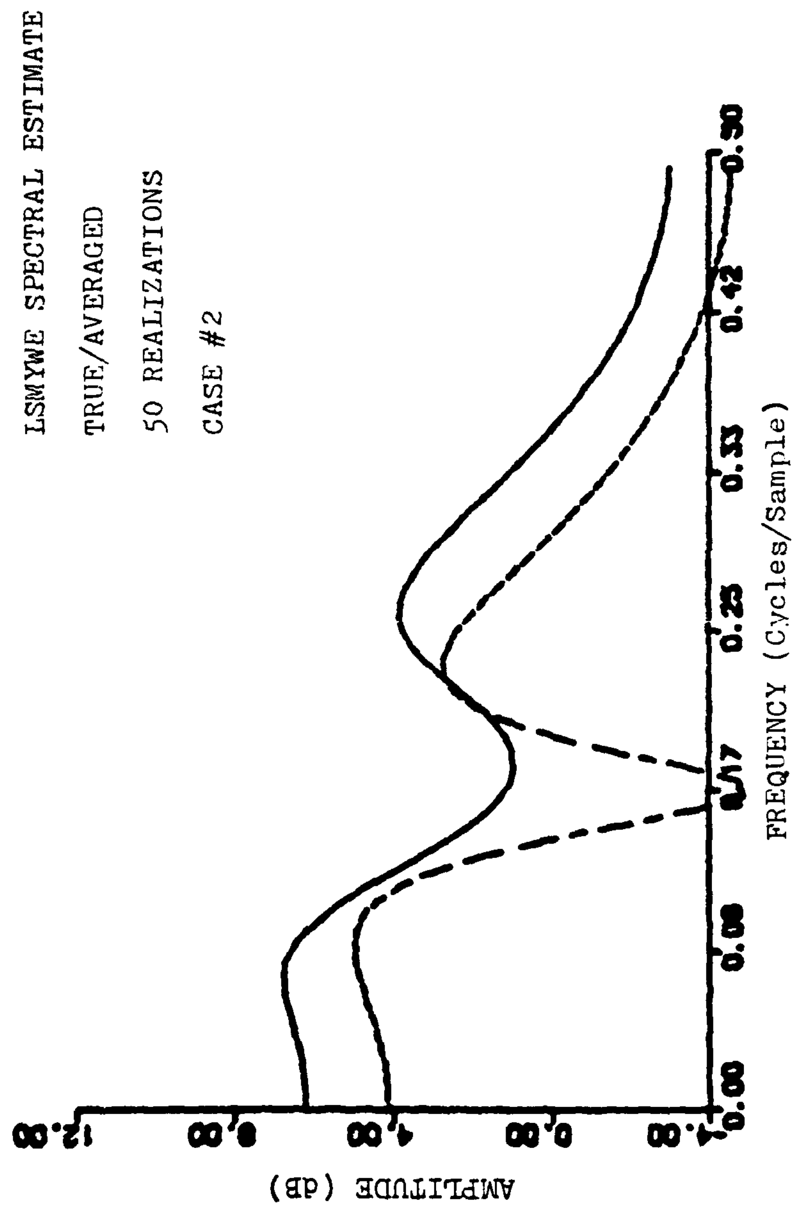


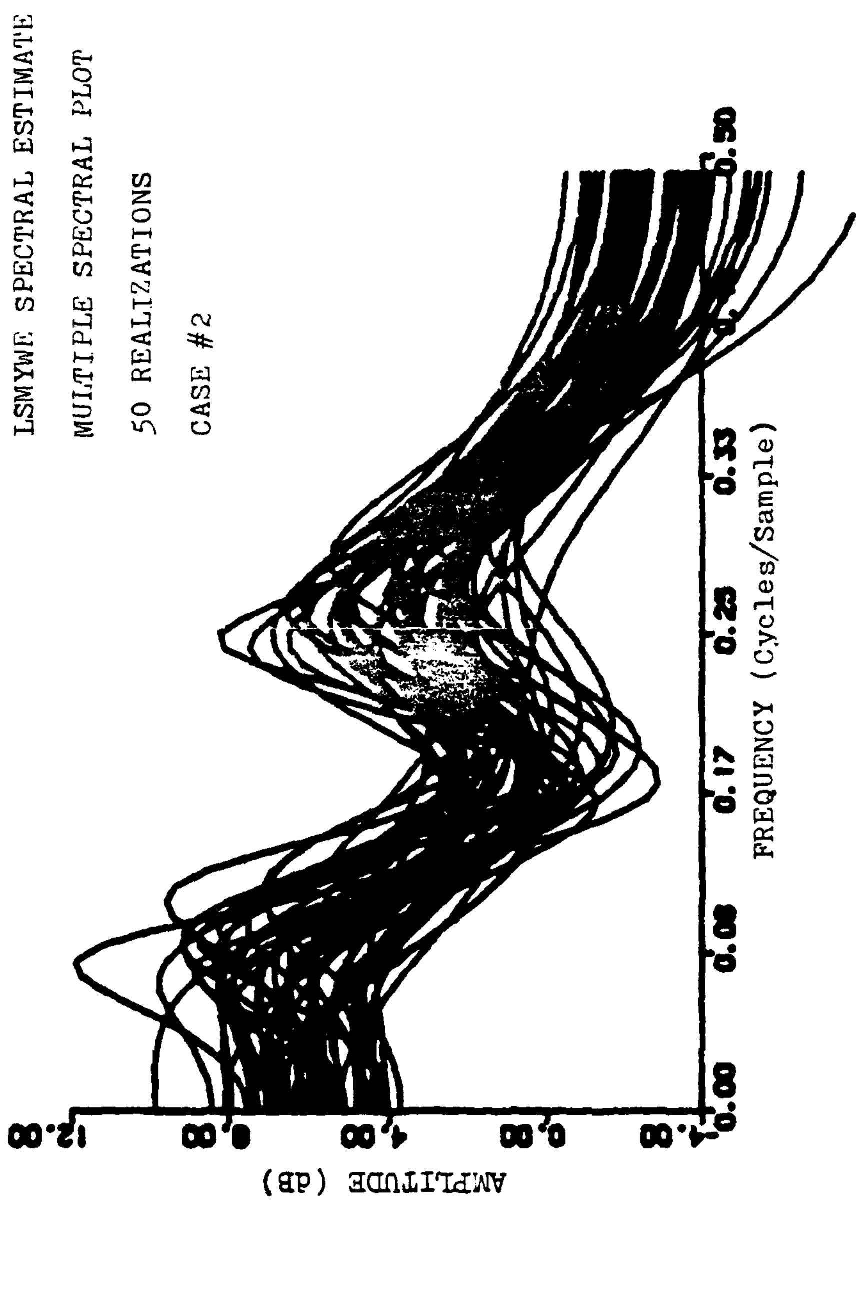
LSMYWE SPECTRAL ESTIMATE

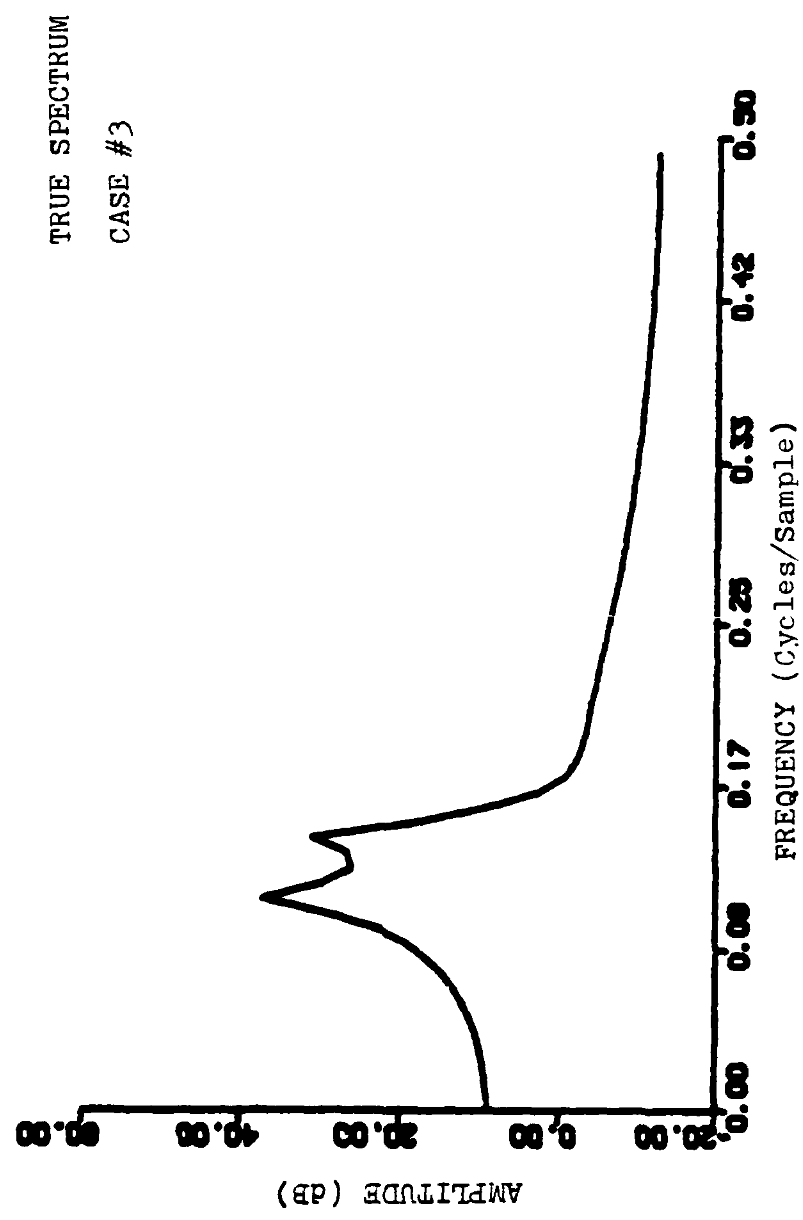
POLE/ZERO PLOT

ONE REALIZATION

CASE #2

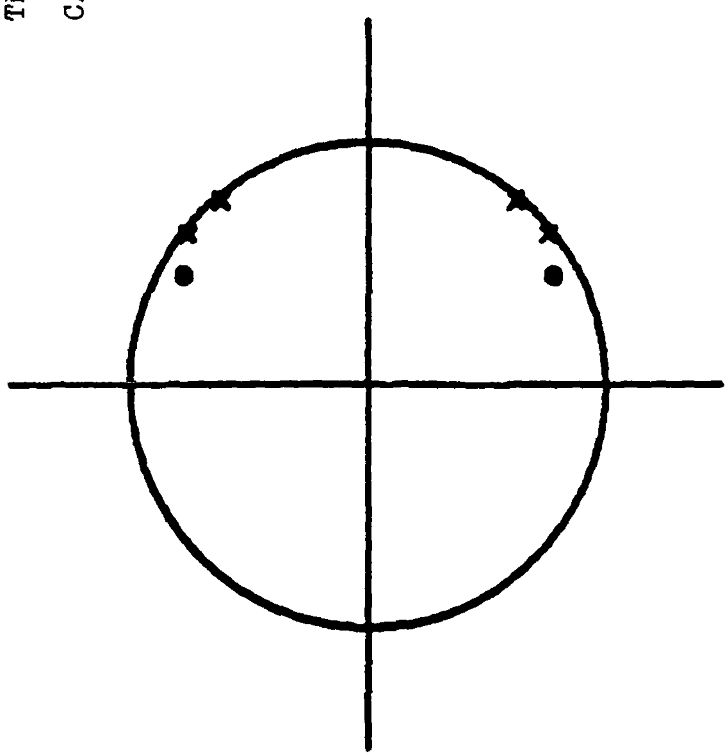

LSMYWE SPECTRAL ESTIMATE
TRUE/AVERAGED
50 REALIZATIONS
CASE #2
AMPLITUDE (dB)
-4.00
0.00
4.00
8.00
12.00
FREQUENCY (Cycles/Sample)
0.00
0.08
0.17
0.25
0.33
0.42
0.50


LSMYWE SPECTRAL ESTIMATE
MULTIPLE SPECTRAL PLOT
50 REALIZATIONS
CASE #2
AMPLITUDE (dB)
12.00
8.00
4.00
0.00
-4.00
FREQUENCY (Cycles/Sample)
0.00
0.08
0.17
0.25
0.33
0.50


TRUE SPECTRUM
CASE #3
FREQUENCY (Cycles/Sample)
0.00
0.08
0.17
0.25
0.33
0.42
0.50
AMPLITUDE (dB)
-20.00
0.00
20.00
40.00
60.00

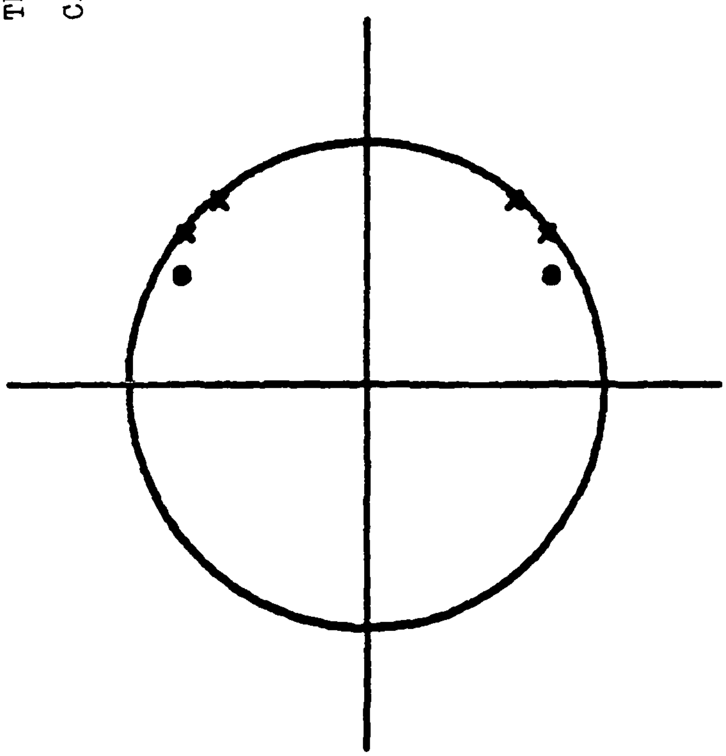
TRUE POLE/ZERO PLOT

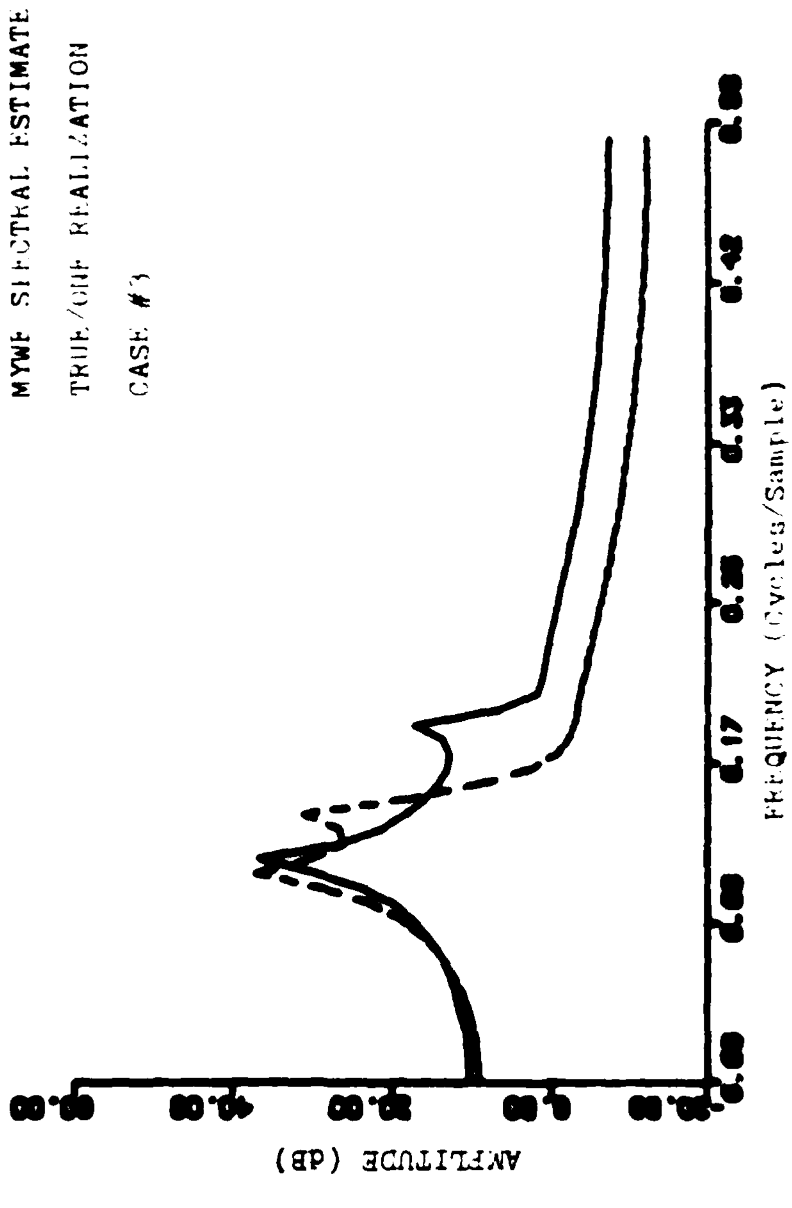
CASE #3



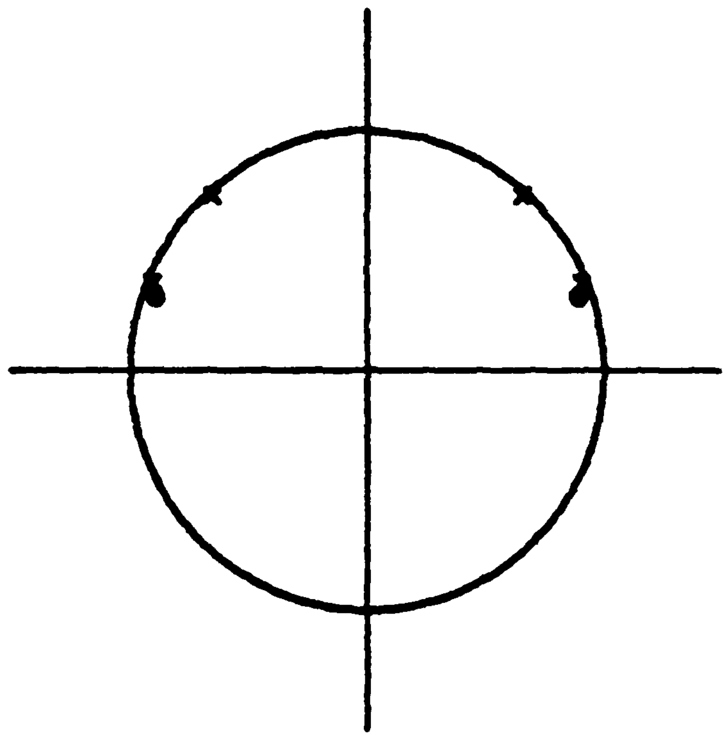
TRUE POLE/ZERO PLOT

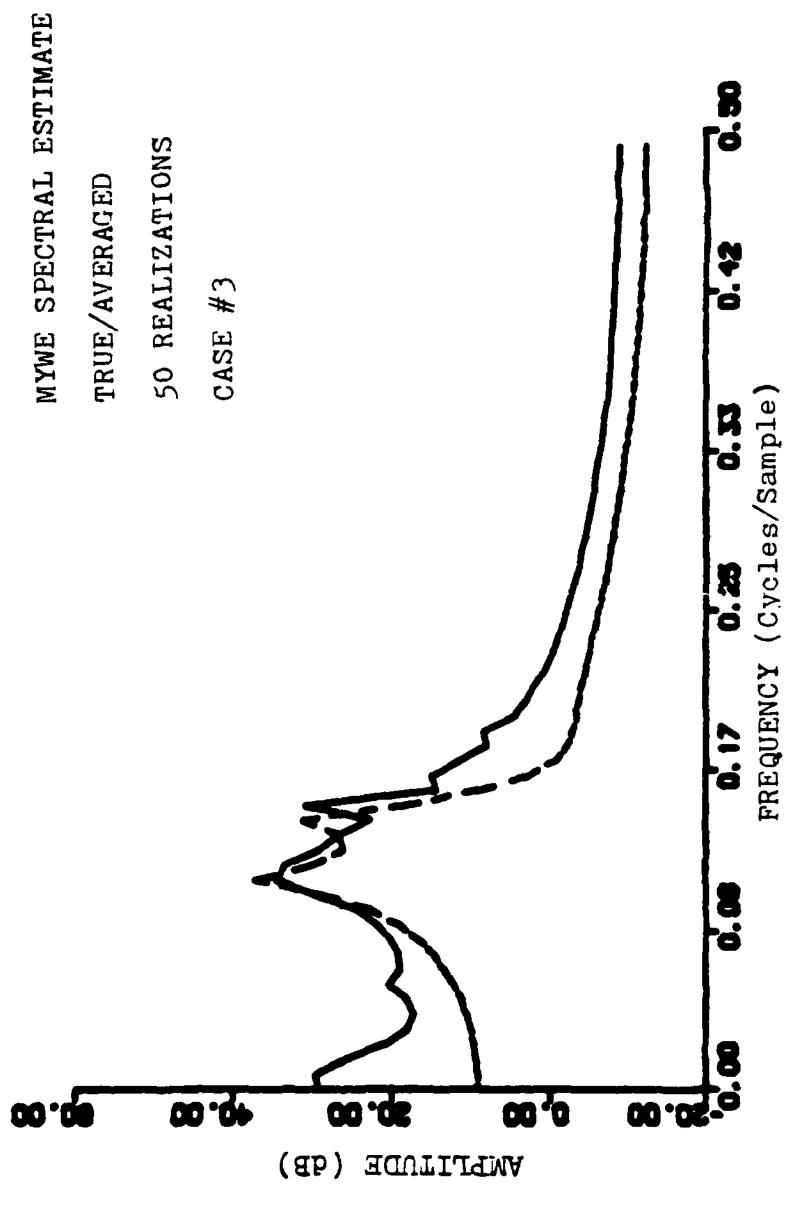
CASE #3

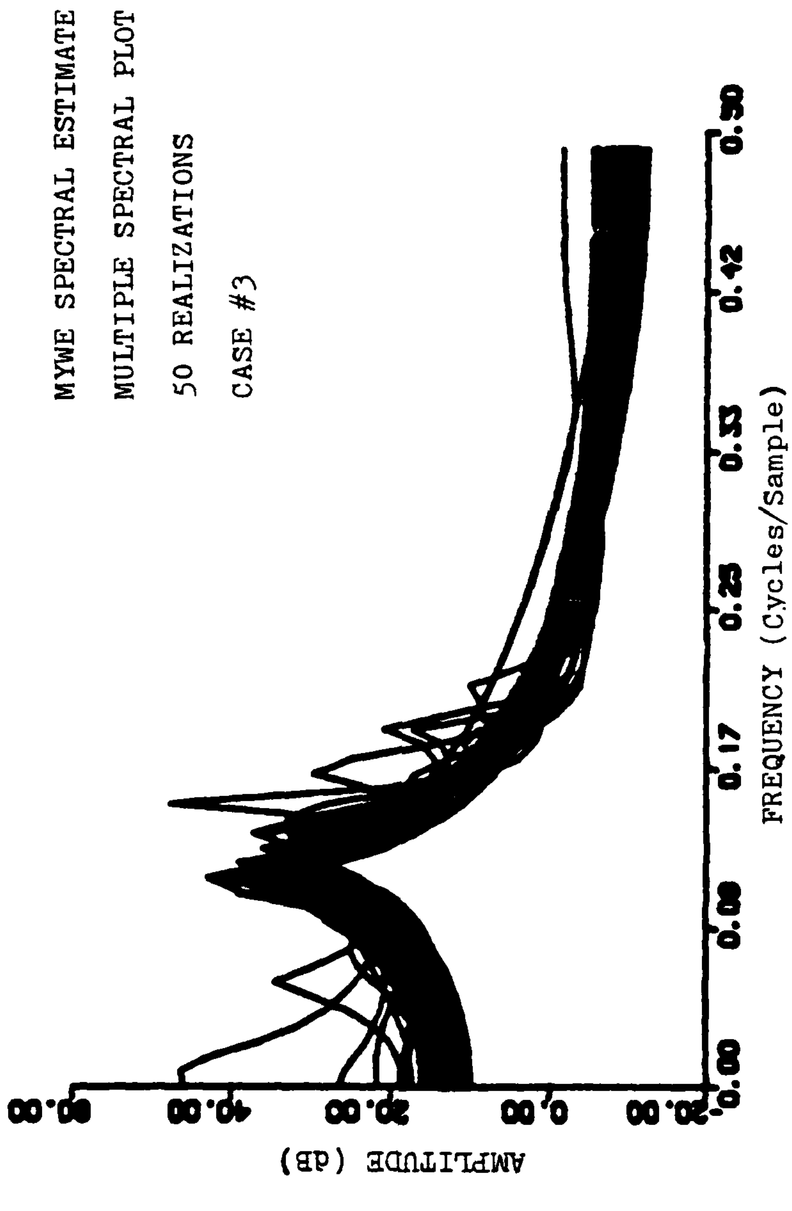


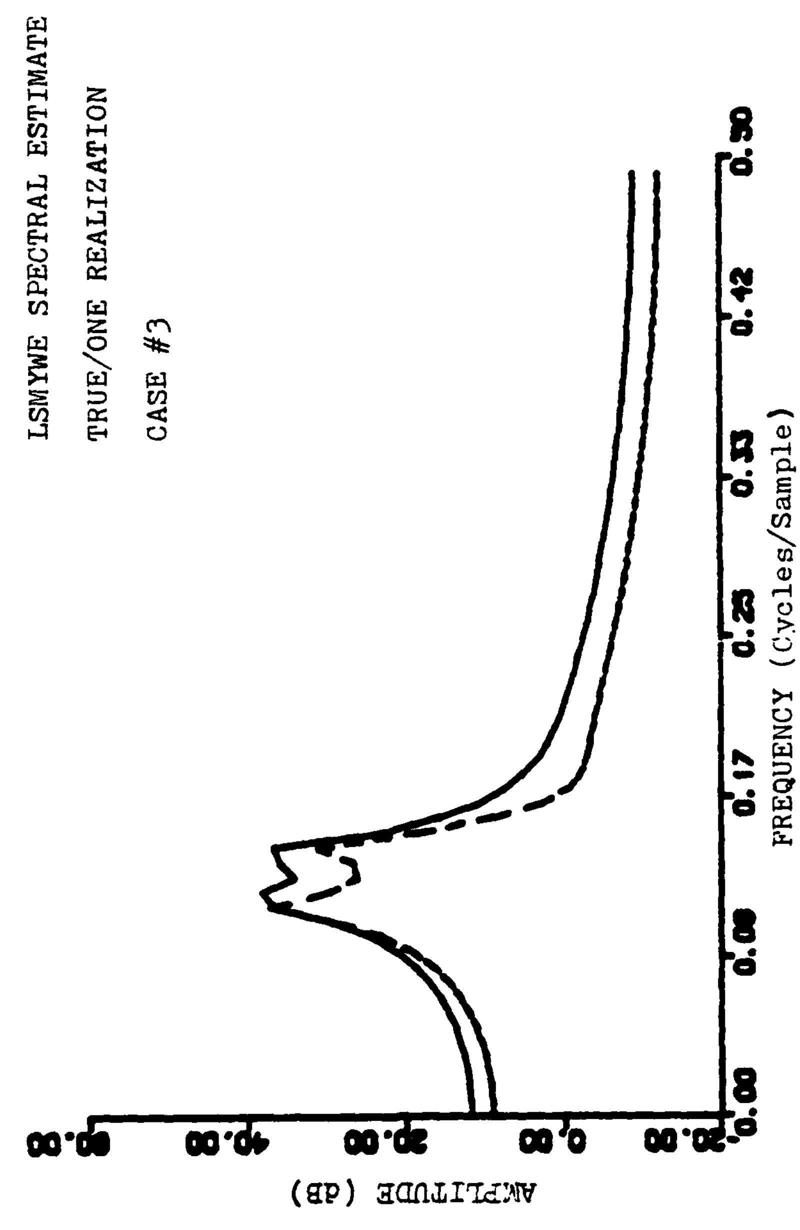

MYWE SPECTRAL ESTIMATE
TRUE/ONE REALIZATION
CASE #3
FREQUENCY (Cycles/Sample)
AMPLITUDE (dB)

MYWE SPECTRAL ESTIMATE
POLE/ZERO PLOT
ONE REALIZATION
CASE #3




MYWE SPECTRAL ESTIMATE
TRUE/AVERAGED
50 REALIZATIONS
CASE #3
AMPLITUDE (dB)
-20.00
0.00
20.00
40.00
60.00
FREQUENCY (Cycles/Sample)
0.00
0.08
0.17
0.25
0.33
0.42
0.50


MYWE SPECTRAL ESTIMATE
MULTIPLE SPECTRAL PLOT
50 REALIZATIONS
CASE #3
AMPLITUDE (dB)
60.00
40.00
20.00
0.00
-20.00
FREQUENCY (Cycles/Sample)
0.00
0.08
0.17
0.25
0.33
0.42
0.50

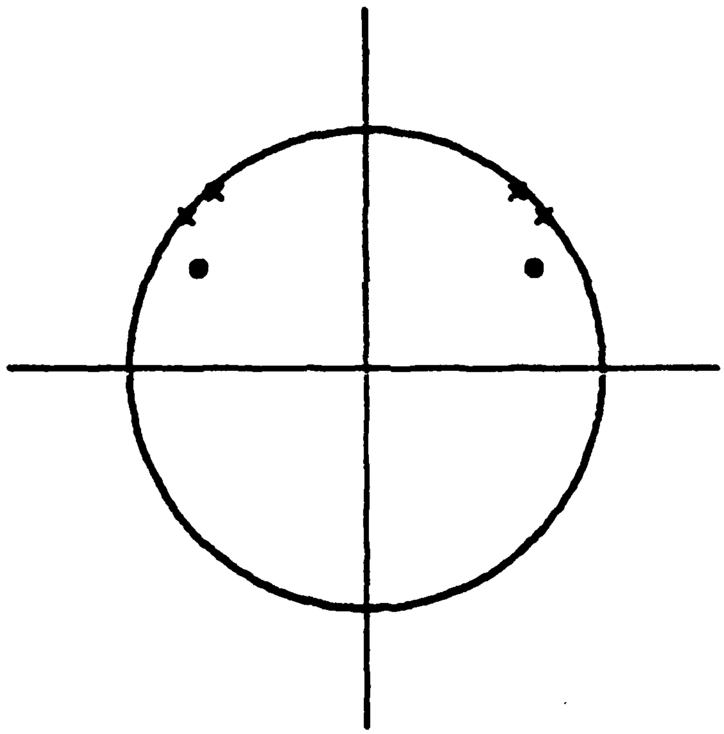

LSMYWE SPECTRAL ESTIMATE
TRUE/ONE REALIZATION
CASE #3
AMPLITUDE (dB)
60.00
40.00
20.00
0.00
-20.00
FREQUENCY (Cycles/Sample)
0.00
0.08
0.17
0.25
0.33
0.42
0.50

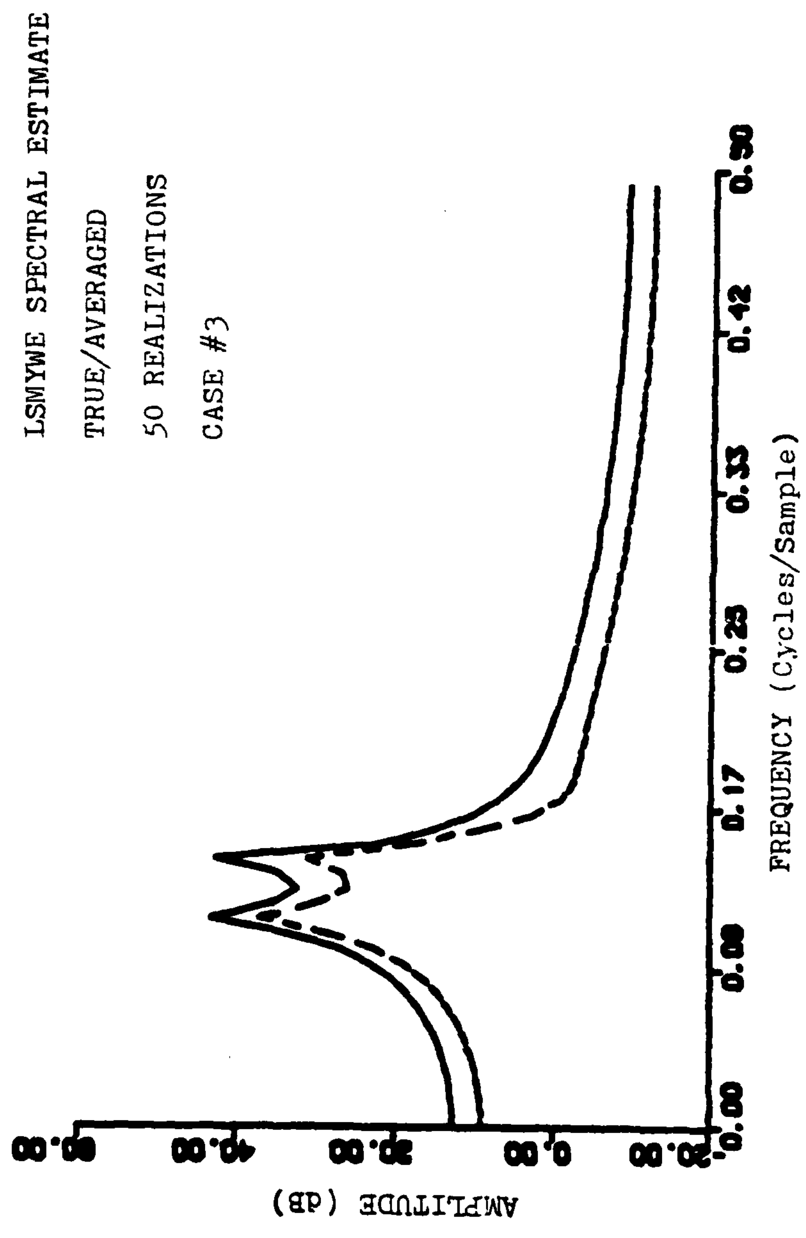
LSMYWE SPECTRAL ESTIMATE

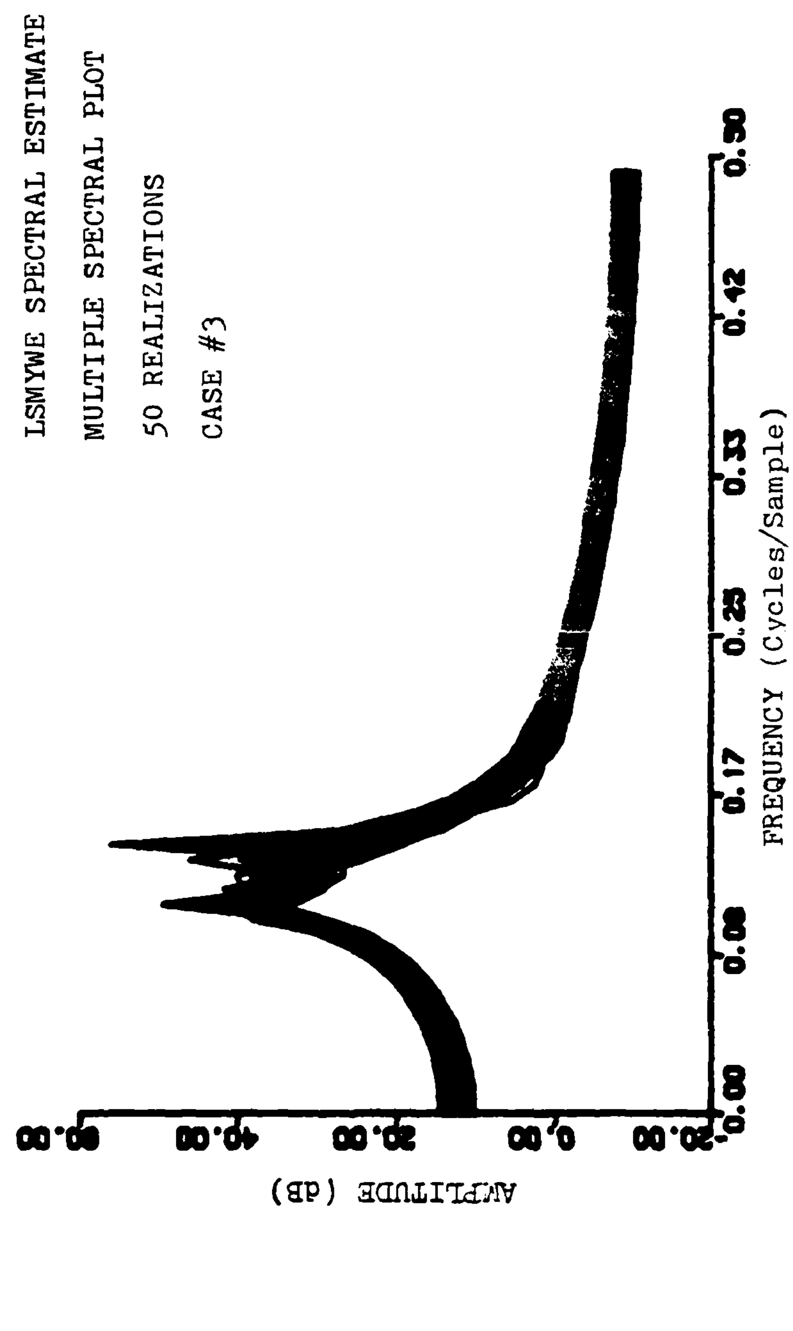
POLE/ZERO PLOT

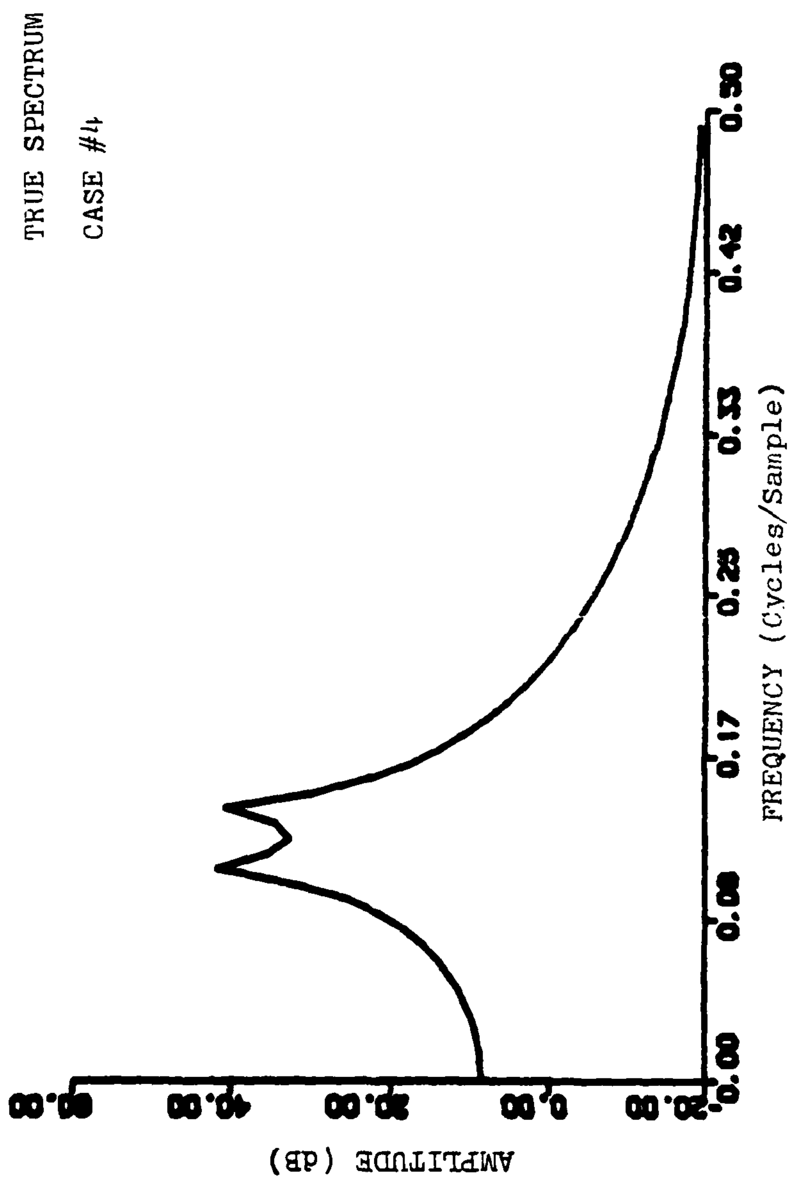
ONE REALIZATION

CASE #3



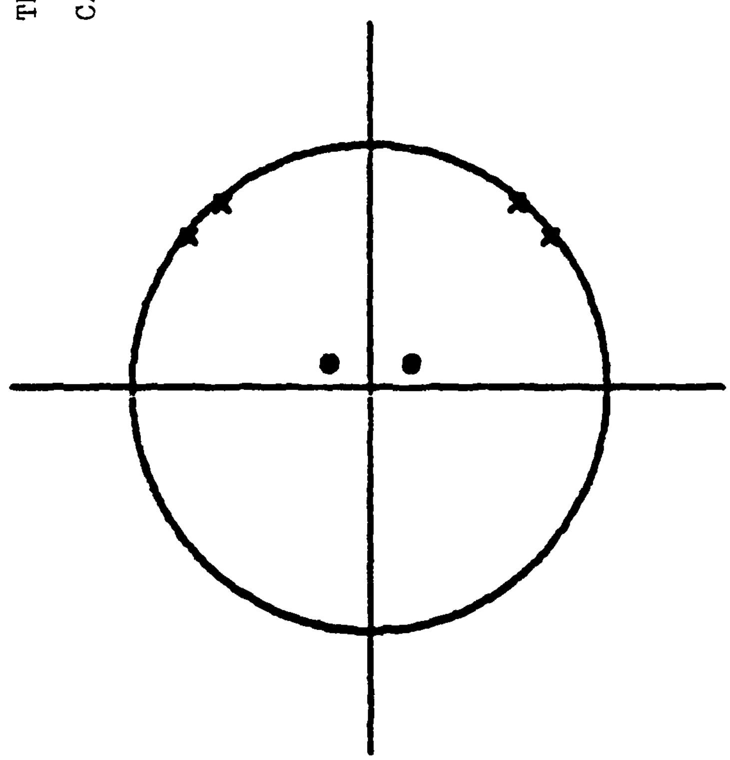

LSMYWE SPECTRAL ESTIMATE
TRUE/AVERAGED
50 REALIZATIONS
CASE #3
AMPLITUDE (dB)
-20.00
0.00
20.00
40.00
60.00
0.00
0.08
0.17
0.25
0.33
0.42
0.50
FREQUENCY (Cycles/Sample)

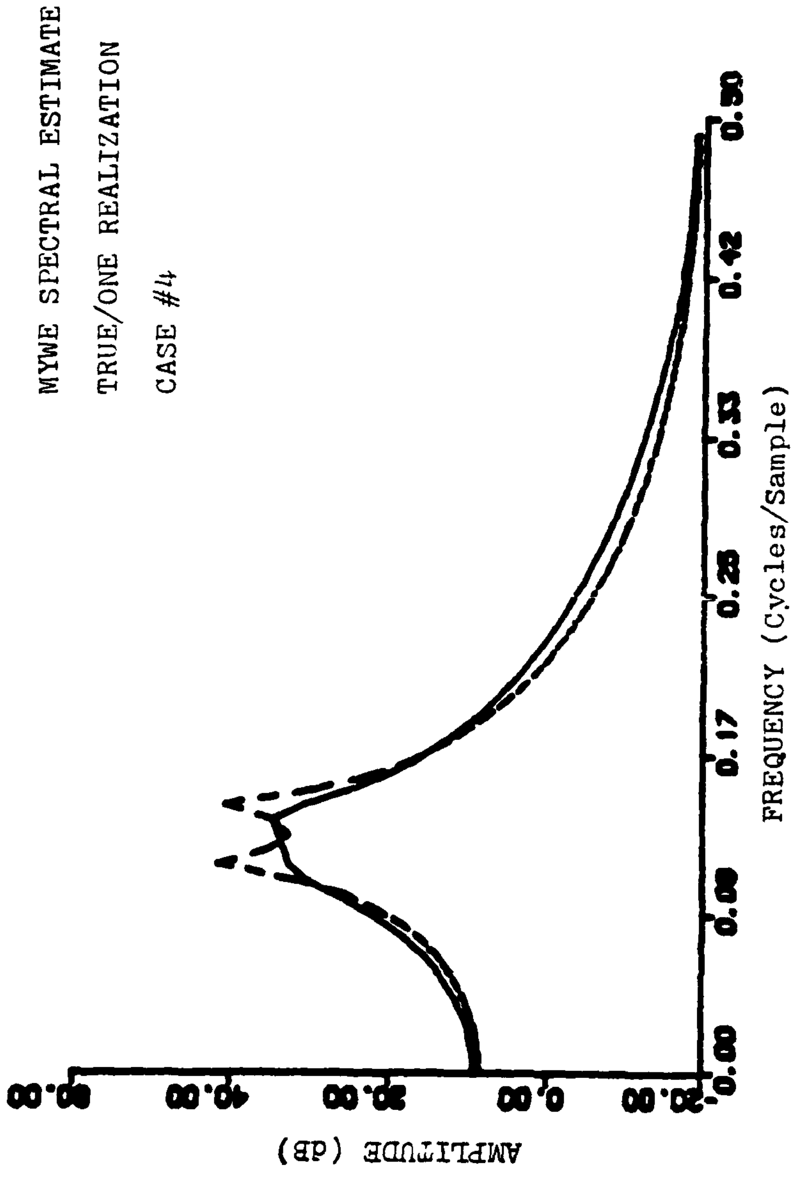

LSMYWE SPECTRAL ESTIMATE
MULTIPLE SPECTRAL PLOT
50 REALIZATIONS
CASE #3
AMPLITUDE (dB)
-20.00
0.00
20.00
40.00
60.00
0.00
0.08
0.17
0.25
0.33
0.42
0.50
FREQUENCY (Cycles/Sample)


TRUE SPECTRUM
CASE #4
FREQUENCY (Cycles/Sample)
AMPLITUDE (dB)
0.00
0.08
0.17
0.25
0.33
0.42
0.50
-20.00
0.00
20.00
40.00
60.00

TRUE POLE/ZERO PLOT

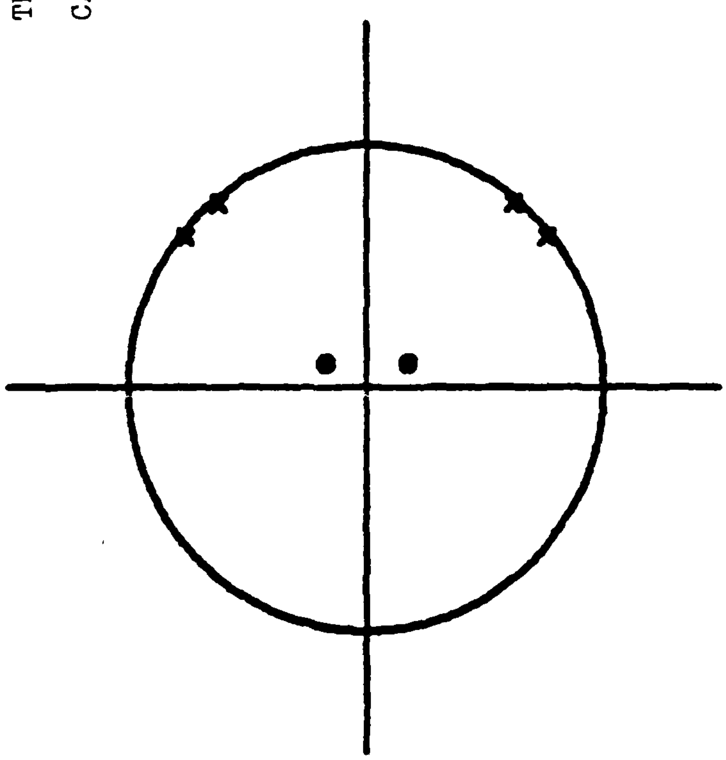
CASE #4




MYWE SPECTRAL ESTIMATE
TRUE/ONE REALIZATION
CASE #4
AMPLITUDE (dB)
-20.00
0.00
20.00
40.00
60.00
0.00
0.08
0.17
0.25
0.33
0.42
0.50
FREQUENCY (Cycles/Sample)

TRUE POLE/ZERO PLOT

CASE #4

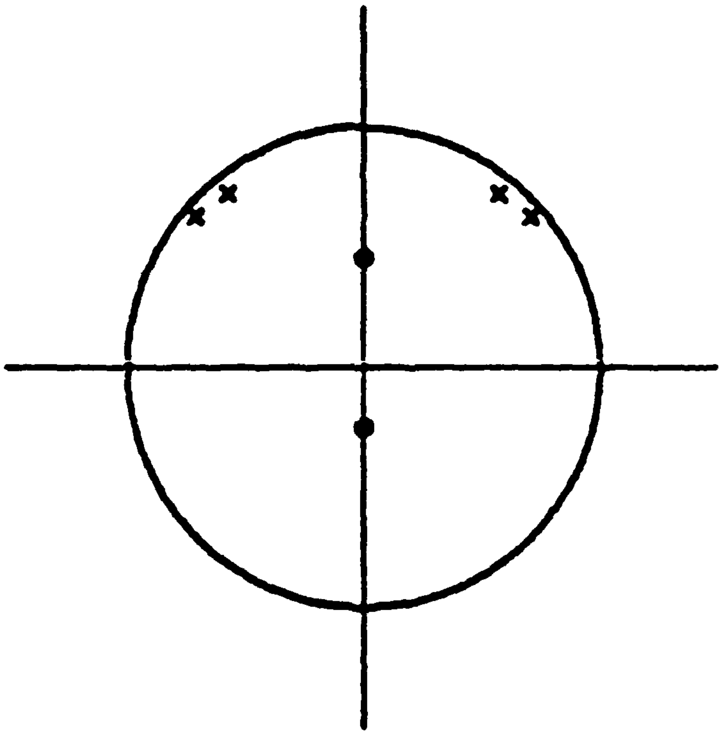


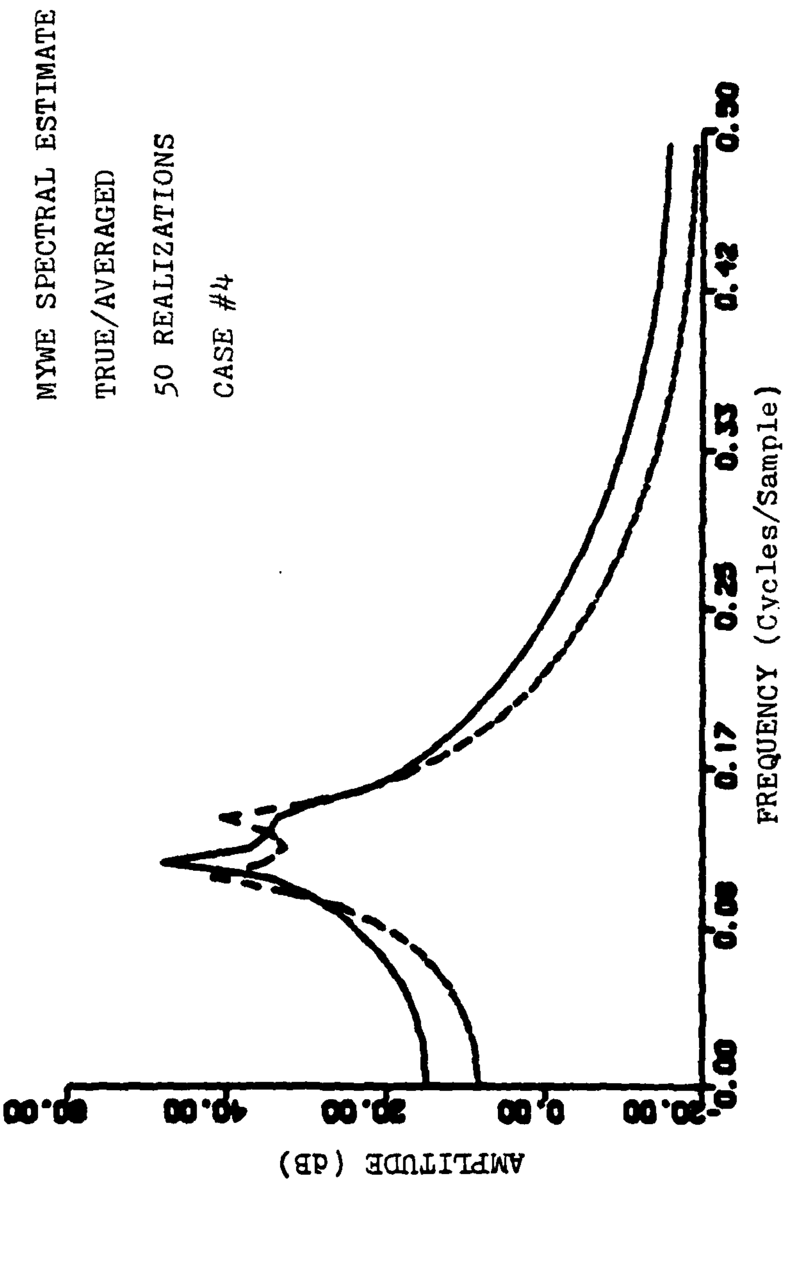
MYWE SPECTRAL ESTIMATE

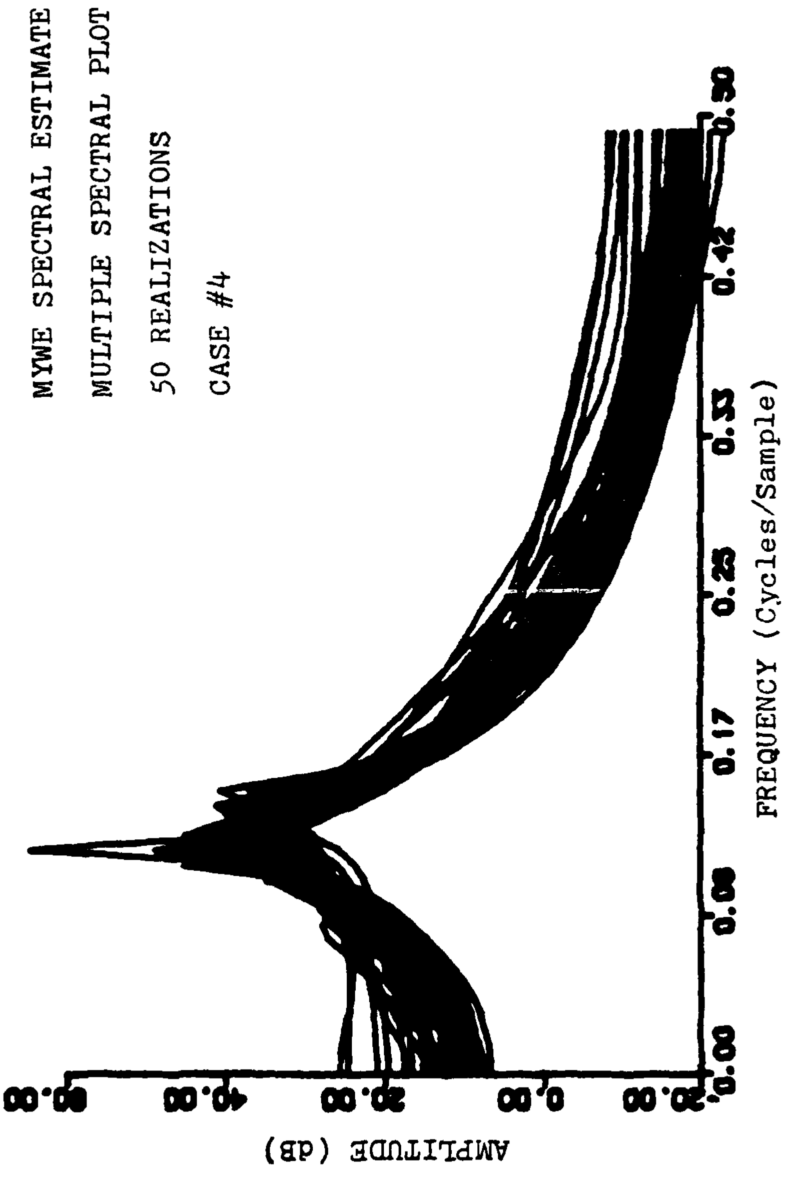
POLE/ZERO PLOT

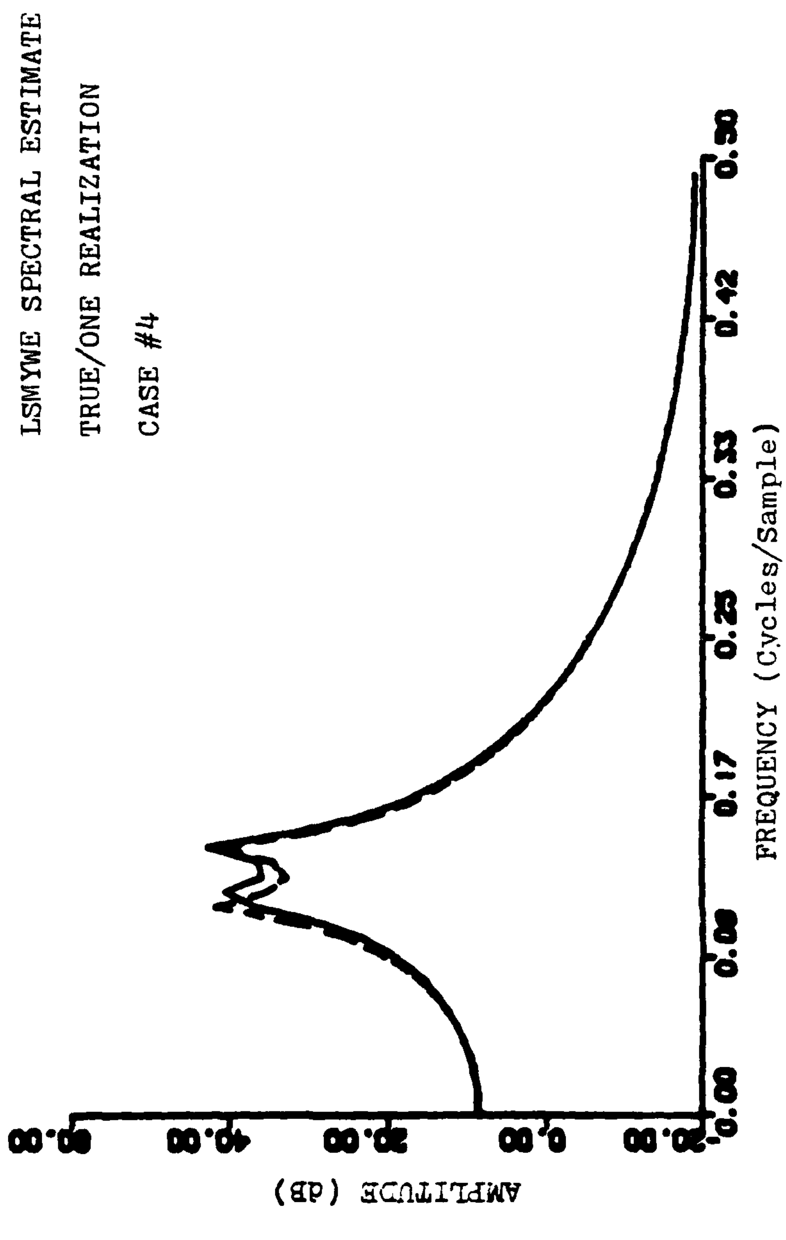
ONE REALIZATION

CASE #4




MYWE SPECTRAL ESTIMATE
TRUE/AVERAGED
50 REALIZATIONS
CASE #4
AMPLITUDE (dB)
60.00
40.00
20.00
0.00
-20.00
FREQUENCY (Cycles/Sample)
0.00
0.08
0.17
0.25
0.33
0.42
0.50


MYWE SPECTRAL ESTIMATE
MULTIPLE SPECTRAL PLOT
50 REALIZATIONS
CASE #4
AMPLITUDE (dB)
60.00
40.00
20.00
0.00
-20.00
0.00
0.08
0.17
0.25
0.33
0.42
0.50
FREQUENCY (Cycles/Sample)

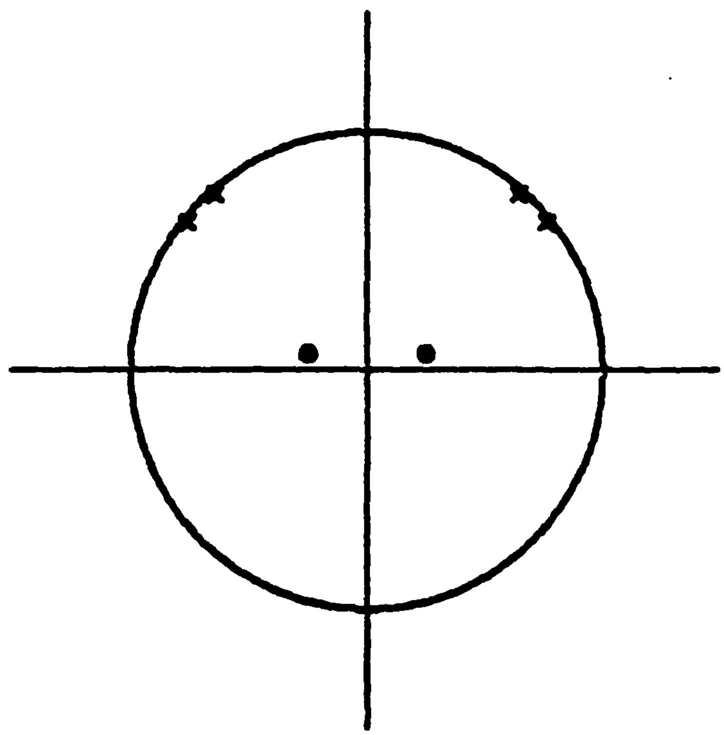

LSMYWE SPECTRAL ESTIMATE
TRUE/ONE REALIZATION
CASE #4
AMPLITUDE (dB)
60.00
40.00
20.00
0.00
-20.00
0.00
0.08
0.17
0.25
0.33
0.42
0.50
FREQUENCY (Cycles/Sample)

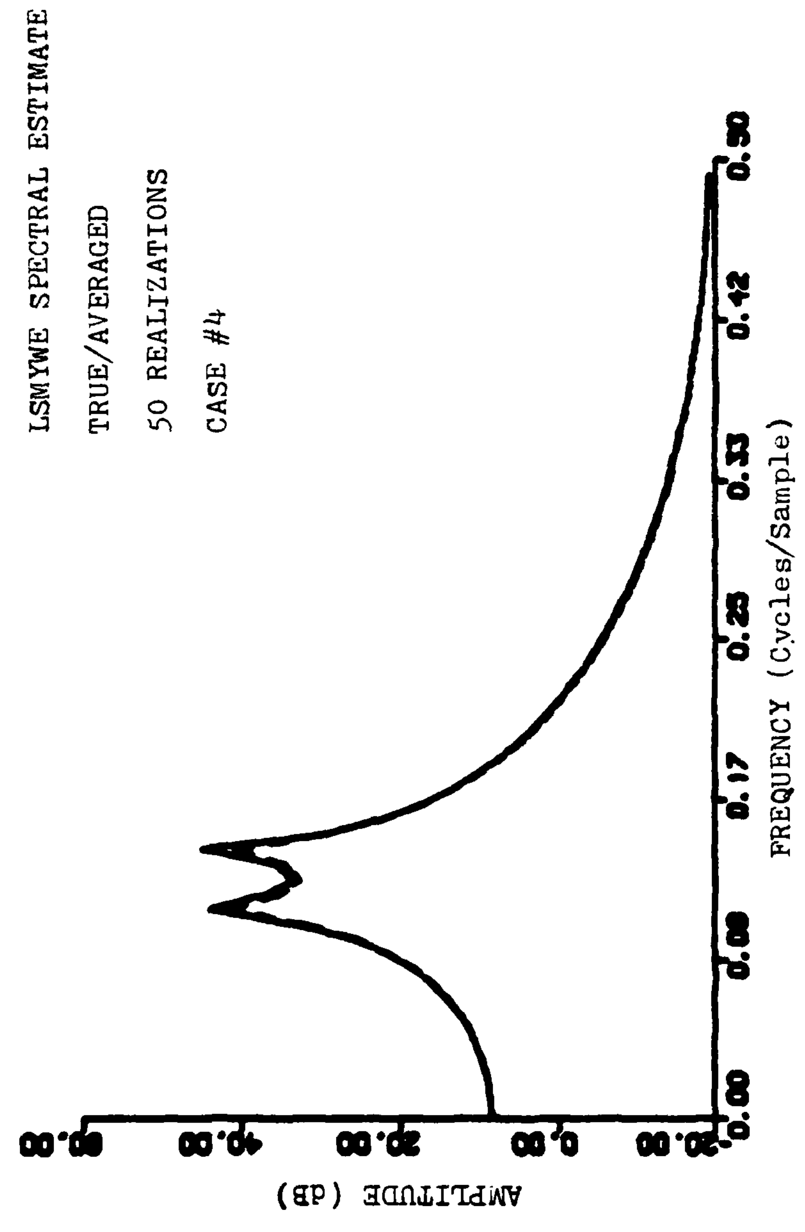
LSMYWE SPECTRAL ESTIMATE

POLE/ZERO PLOT

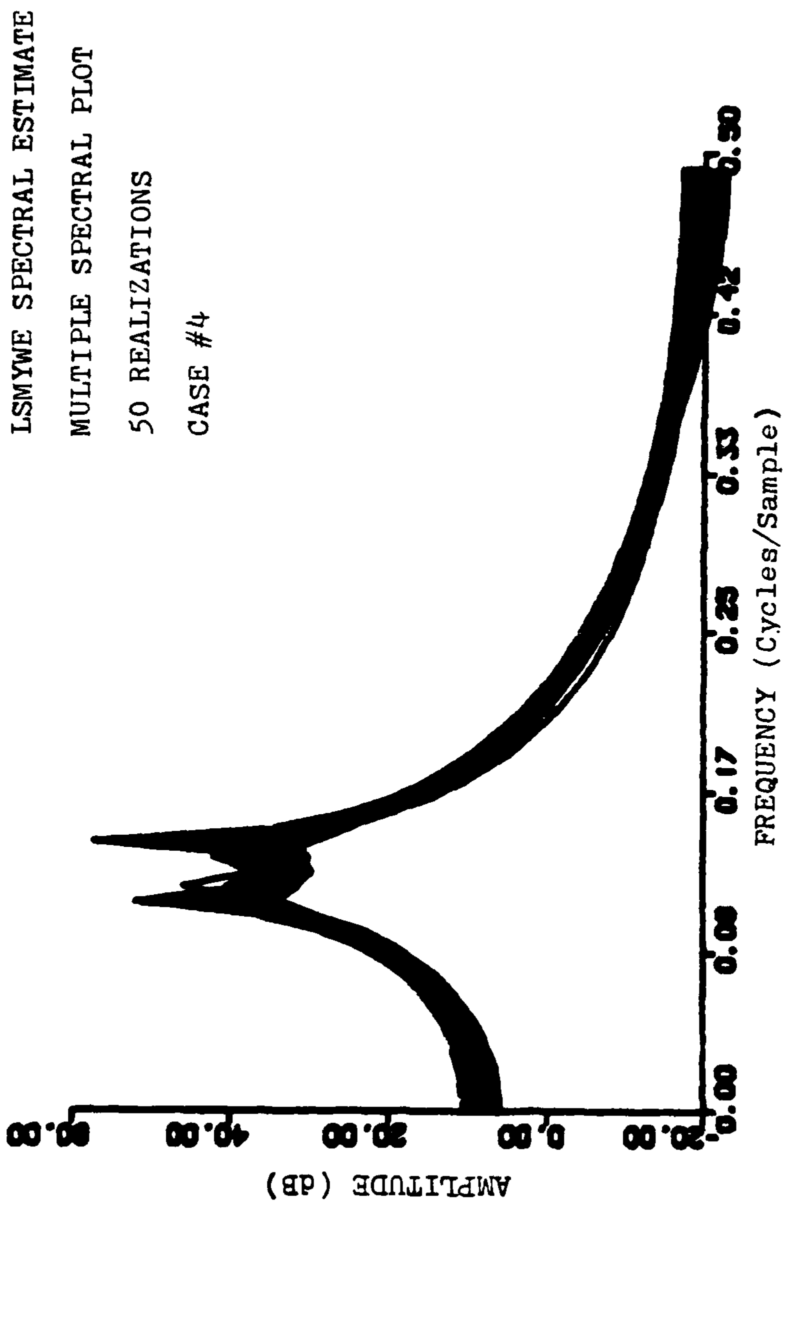
ONE REALIZATION

CASE #4





LSMYWE SPECTRAL ESTIMATE
TRUE/AVERAGED
50 REALIZATIONS
CASE #4
AMPLITUDE (dB)
60.00
40.00
20.00
0.00
-20.00
0.00
0.08
0.17
0.25
0.33
0.42
0.50
FREQUENCY (Cycles/Sample)


LSMYWE SPECTRAL ESTIMATE
MULTIPLE SPECTRAL PLOT
50 REALIZATIONS
CASE #4
AMPLITUDE (dB)
60.00
40.00
20.00
0.00
-20.00
FREQUENCY (Cycles/Sample)
0.00
0.08
0.17
0.25
0.33
0.42
0.50

APPENDIX B

SAMPLE COMPUTER PROGRAMS

```
 PIZZA        JOB (NYF[illegible]),'PIZZ',TIME=(2,30)
  EXEC FORTVCG,REGION.GO=1500K,LIB1=IMSLD,LIB2=CPLOT
//FORT.SYSIN DD *
C       THIS PROGRAM COMPUTES THE MYW SPECTRAL ESTIMATE
        DIMENSION W(20000)
        REAL*8 H(0:2000),A(0:30),B(0:30)
        REAL*8 X(0:20000),Y(0:20000),X3(0:20000)
        REAL*8 X2(0:2000),A2(0:30),Y3(0:2000)
        COMPLEX*16 R(0:500),X1(0:5[illegible]),A1(100,20),R1(100)
        COMPLEX*16 WA(10200),WK(10[illegible])
        COMPLEX*16 C1(100),C(20,20)
        REAL*8 XF(300),AL(100,100),POW(100),X7(300)
        REAL*8 AYW(100,100),R7(101),PYW(100)
C************************************************************
C       READ THE INPUT DATA AND PARAMETERS
C       NUM=NUMBER OF RUNS REQUESTED
C       N=LENGTH OF THE IMPULSE RESPONSE REQUESTED
C       IP=ORDER OF THE AUTO-REGRESSIVE PARAMETERS
C       IQ=ORDER OF THE MOVING-AVERAGE PARAMETERS
C       A(I)=THE AUTO-REGRESSIVE COEFFICIENTS
C       B(I)=THE MOVING AVERAGE COEFFICIENTS
        READ(1,4),N,NUM
        WRITE(6,4),N,NUM
    4   FORMAT(I3,I2)
        READ(1,2),IP,IQ
        WRITE(6,2),IP,IQ
    2   FORMAT(2I2)
        DO 15 K=1,IP
        READ(1,3),A(K)
    3   FORMAT(F11.7)
   15   CONTINUE
        DO 16 J=1,IQ
        READ(1,3),B(J)
   16   CONTINUE
C************************************************************
C       GENERATE GAUSSIAN WHITE NOISE
        NPTS=16000
        ISEED=11111
        VAR=1.
        CALL WGN(W,NPTS,VAR,ISEED)
C       STORE THE WHITE NOISE IN X(I)
        DO 10 I=0,NPTS-1
        X(I)=W(I+1)
   10   CONTINUE
C************************************************************
        B(0)=1.0
C       GENERATE THE IMPULSE RESPONSE
        CALL IMPULS(N,H,A,B,IP,IQ)
  100   FORMAT (4(3X,F9.4))
C       GENERATE ARMA TIME SERIES
        LX=NPTS
        LH=N
        LY=NPTS+N
        CALL CONVOL(H,LH,X,LX,Y,LY)
C       FILTER THE NON-STATIONARY DATA AND SAVE TIME SERIES
C       NS = LENGTH OF THE NON-STATIONARY VALUES OF H(N)
        NS = 1000
```

```
      DO 17 I=0,LY-NS-1
      X3(I)=Y(I+NS)
   17 CONTINUE
C     X1(I) = ARMA STATIONARY TIME SERIES (COMPLEX)
C     X2(I) = ARMA STATIONARY TIME SERIES (REAL)
C     IFL = NUMBER OF STATIONARY POINTS REQUESTED
C     IFL MUST BE .LT. Y(I+NS)
      WRITE(2,200),NUM
      NUM1=0
      DO 800 KK2=1,NUM
      IFL=256
      DO 20 I=0,IFL-1
      X1(I)=X3(I+NUM1)
      X2(I)=X3(I+NUM1)
   20 CONTINUE
C     COMPUTE THE AUTO-CORRELATION FUNCTION
C     SET LR=LENGTH OF AUTO-CORRELATION LAGS REQUESTED
C     FOR MYWE LR=IP+IQ
C
      LR=25
      LN=IFL
      CALL AUTO(X1,R,LR,LN)
C     GENERATE THE MODIFIED YULE-WALKER EQUATIONS
      CALL MODY(R,IP,IQ,A1,R1,IP)
C     INVERT THE MYWE AND SOLVE
      CALL LEQ2C(A1,IP,100,R1,1,100,0,WA,WK,IER)
      WRITE(6,200)IER
  200 FORMAT(I2)
C     SAVE THE AR PARAMETERS ESTIMATES
      DO 45 I=1,IP
      A2(I)=R1(I)
   45 CONTINUE
C     FILTER THE AR PARAMETERS
      NPTS1=IFL
      WRITE(6,100),(A2(K),K=1,IP)
      CALL ARFIL(X2,IP,A2,Y3,LY3,NPTS1)
C     COMPUTE THE MOVING AVERAGE PARAMETERS
      CALL DURBIN(Y3,LY3,IQ,80,AL,POWO1)
      WRITE(6,100),(AL(I,2),I=1,IQ)
      WRITE(6,150),POWO1
  150 FORMAT(F11.7)
      WRITE(2,700),(A2(K),K=1,IP),(AL(I,IQ),I=1,IQ),POWO1
  700 FORMAT(21F11.7)
      WRITE(6,200),KK2
      NUM1=NUM1+IFL
  800 CONTINUE
      STOP
      END
```

```
//PIZZA        JOB (NYF101),'PIZZ',TIME=(2,30)
// EXEC FORTVCG,REGION.GO=1500K,LIB1=IMSLD,LIB2=CPLOT
//FORT.SYSIN DD *
C       THIS PROGRAMS COMPUTES THE LSMYW SPECTRAL ESTIMATE
        DIMENSION W(20000)
        REAL*8 H(0:2000),A(0:30),B(0:30)
        REAL*8 X(0:20000),Y(0:20000),X3(0:20000)
        REAL*8 X2(0:2000),A2(0:30),Y3(0:2000)
        COMPLEX*16 R(0:500),X1(0:500),A1(100,20),R1(100)
        COMPLEX*16 WA(10200),WK(100)
        COMPLEX*16 C1(100),C(20,20)
        REAL*8 XF(300),AL(100,100),POW(100),X7(300)
        REAL*8 AYW(100,100),R7(101),PYW(100)
C**************************************************************
C       READ THE INPUT DATA AND PARAMETERS
C       NUM=NUMBER OF RUNS REQUESTED
C       N=LENGTH OF THE IMPULSE RESPONSE REQUESTED
C       IP=ORDER OF THE AUTO-REGRESSIVE PARAMETERS
C       IQ=ORDER OF THE MOVING-AVERAGE PARAMETERS
C       A(I)=THE AUTO-REGRESSIVE COEFFICIENTS
C       B(I)=THE MOVING AVERAGE COEFFICIENTS
        READ(1,4),N,NUM
  4     FORMAT(I3,I2)
        READ(1,2),IP,IQ
  2     FORMAT(2I2)
        DO 15 K=1,IP
        READ(1,3),A(K)
  3     FORMAT(F11.7)
  15    CONTINUE
        DO 16 J=1,IQ
        READ(1,3),B(J)
  16    CONTINUE
C**************************************************************
C       GENERATE GAUSSIAN WHITE NOISE
        NPTS=16000
        ISEED=11111
        VAR=1.
        CALL WGN(W,NPTS,VAR,ISEED)
C       STORE THE WHITE NOISE IN X(I)
        DO 10 I=0,NPTS-1
        X(I)=W(I+1)
 10     CONTINUE
C**************************************************************
        B(0)=1.0
C       GENERATE THE IMPULSE RESPONSE
        CALL IMPULS(N,H,A,B,IP,IQ)
 100    FORMAT (4(3X,F9.4))
C       GENERATE ARMA TIME SERIES
        LX=NPTS
        LH=N
        LY=NPTS+N
        CALL CONVOL(H,LH,X,LX,Y,LY)
C       FILTER THE NON-STATIONARY DATA AND SAVE TIME SERIES
C       NS = LENGTH OF THE NON-STATIONARY VALUES OF H(N)
        NS = 1000
```

```
      DO 17 I=0,LY-NS-1
      X3(I)=Y(I+NS)
   17 CONTINUE
C     X1(I) = ARMA STATIONARY TIME SERIES (COMPLEX)
C     X2(I) = ARMA STATIONARY TIME SERIES (REAL)
C     IFL = NUMBER OF STATIONARY POINTS REQUESTED
C     IFL MUST BE .LT. (LY-NS-1)
      WRITE(2,200),NUM
      NUM1=0
      DO 800 KK2=1,NUM
      IFL=256
      DO 20 I=0,IFL-1
      X1(I)=X3(I+NUM1)
      X2(I)=X3(I+NUM1)
   20 CONTINUE
C     COMPUTE THE AUTO-CORRELATION FUNCTION
C     SET LR=LENGTH OF AUTO-CORRELATION LAGS REQUESTED
C     FOR LSMYWE LR.GT.M1
C
      LR=75
      LN=IFL
      M1=50
      CALL AUTO(X1,R,LR,LN)
C     GENERATE THE MODIFIED YULE-WALKER EQUATIONS
      CALL MODY(R,IP,IQ,A1,R1,M1)
      IR=M1-IQ
      CALL INVR(A1,R1,IP,IR,C,C1)
C     INVERT THE MYWE AND SOLVE
      CALL LEQ2C(C,IP,100,C1,1,100,0,WA,WK,IER)
      WRITE(6,200)IER
  200 FORMAT(I2)
C     SAVE THE AR PARAMETERS ESTIMATES
      DO 45 I=1,IP
      A2(I)=C1(I)
   45 CONTINUE
C     FILTER THE AR PARAMETERS
      NPTS1=IFL
      WRITE(6,100),(A2(K),K=1,IP)
      CALL ARFIL(X2,IP,A2,Y3,LY3,NPTS1)
C     COMPUTE THE MOVING AVERAGE PARAMETERS
      CALL DURBIN(Y3,LY3,IQ,80,AL,POW01)
      WRITE(6,100),(AL(I,2),I=1,IQ)
      WRITE(6,150),POW01
  150 FORMAT(F10.7)
      WRITE(2,700),(A2(K),K=1,IP),(AL(I,IQ),I=1,IQ),POW01
  700 FORMAT(21F11.7)
      WRITE(6,200),KK2
      NUM1=NUM1+IFL
  800 CONTINUE
      STOP
      END
```

```
C***********************************************************
C      THIS PROGRAM GENERATES WHITE GAUSSIAN RANDOM NOISE
C
C      INPUT PARAMETERS
C      VAR = THE VARIANCE
C      ISEED = 11111
C      NPTS = NUMBER OF DATA POINTS REQUESTED
C
C      OUTPUT PARAMETERS
C      W(I) = THE OUTPUT TIME SERIES
C
C***********************************************************
C
       SUBROUTINE WGN(W,NPTS,VAR,ISEED)
       DIMENSION W(512)
       M=NPTS
       IF (MOD(NPTS,2).NE.0)M=NPTS+1
       DO 10 I=1,M
       CALL RANDU(ISEED,IY,YFL)
       ISEED=IY
 10    W(I)=YFL
       CALL NORRAN(W,M,0.,VAR)
       RETURN
       END
       SUBROUTINE NORRAN(W,M,RMEAN,VAR)
       DIMENSION W(512)
       PI=4.0*ATAN(1.0)
       L=M/2
       DO 10 I=1,L
       U1=W(2*I-1)
       U2=W(2*I)
       TEMP=SQRT(-2.0*ALOG(U1))
       W(2*I-1)=TEMP*COS(2.0*PI*U2)*SQRT(VAR)+RMEAN
 10    W(2*I)=TEMP*SIN(2.0*PI*U2)*SQRT(VAR)+RMEAN
       RETURN
       END
       SUBROUTINE RANDU(IX,IY,YFL)
       IY=IX*65539
       IF(IY)5,6,6
 5     IY=IY+2147483647+1
 6     YFL=IY
       YFL=YFL*.4656613E-9
       RETURN
       END
```

```
C**********************************************************
C      THIS PROGRAM CALCULATES THE IMPULSE RESPONSE OF AN
C      AUTO-REGRESSIVE/MOVING AVERAGE FILTER TRANSFER FUNCTION
C
C      INPUT PARAMETERS
C      A(I) = THE AUTO-REGESSIVE COEFFICIENTS
C      B(I) = THE MOVING AVERAGE COEFFICIENTS
C      IP = THE ORDER OF THE AUTO-REGRESSIVE COEFFICIENTS
C      IQ = THE ORDER OF THE MOVING AVERAGE COEFFICIENTS
C      N = NUMBER OF FILTER RESPONSES REQUESTED
C
C      OUTPUT PARAMETERS
C      H(N) = OUTPUT DATA VECTOR
C**********************************************************
C
       SUBROUTINE IMPULS(N,H,A,B,IP,IQ)
       REAL*8 H(0:2000),A(0:30),B(0:30)
       H(0)=B(0)
       DO 10 I=1,N-1
       IC=I
       H(I)=0D0
       IF (I.GT.IP) IC=IP
       DO 20 J=1,IC
       H(I)=H(I)-A(J)*H(I-J)
   20  CONTINUE
       IF (I.GT.IQ) GO TO 10
       H(I)=H(I)+B(I)
   10  CONTINUE
       RETURN
       END
```

```
C*********************************************************
C      THIS PROGRAM CONVOLVES THE INPUT SIGNEL X(N)
C      WITH THE FILTER IMPULSE RESPONSE H(N)
C
C      INPUT PARAMETERS
C      X(I) = THE INPUT TIME SERIES
C      LX = THE LENGTH OF THE INPUT TIME SERIES
C      H(I) = THE IMPULSE RESPONSE
C      LH = THE LENGTH OF THE IMPULSE RESPONSE
C
C      OUTPUT PARAMETERS
C      Y(I) = THE OUTPUT TIME SERIES
C      LY = THE LENGTH OF THE OUTPUT TIME SERIES
C
C      **** LX SHOULD BE GE. LH ****
C
C*********************************************************
C
       SUBROUTINE CONVOL(H,LH,X,LX,Y,LY)
       REAL*8 H(0:2000),X(0:2000),Y(0:2000)
       INTEGER LH,LX,LY
       DO 10 I=0,LY-1
       Y(I)=0D0
       N=MIN0(I,LH-1)
       DO 20 J=0,N
       Y(I)=Y(I)+H(J)*X(I-J)
  20   CONTINUE
  10   CONTINUE
       RETURN
       END
```

```
C*******************************************************************
C       THIS PROGRAM CALCULATES THE UNBIASED AUTO-CORRELATION
C       ESTIMATE OF A COMPLEX FUNCTION
C
C       INPUT PARAMETERS
C       LR = NUMBER OF LAGS CALCULATED
C       X1(I) = INPUT TIME SERIES
C       LN = THE LENGTH OF THE INPUT TIME SERIES
C
C       OUTPUT PARAMETERS
C       R(I) = THE AUTO-CORRELATION ESTIMATES
C*******************************************************************
C
        SUBROUTINE AUTO(X1,R,LR,LN)
        INTEGER LR
        COMPLEX*16 R(0:500),X1(0:500)
        DO 10 K=0,LR-1
        N=LN-1-K
        R(K)=(0D0,0D0)
        DO 20 J=0,N
        R(K)=R(K)+DCONJG(X1(J))*X1(K+J)
  20    CONTINUE
        M=LN-K
        R(K)=R(K)/FLOAT(M)
  10    CONTINUE
        RETURN
        END
```

```
C**********************************************************
C      THIS PROGRAM CALCULATES THE MODIFED YULE-WALKER
C      ESTIMATES BY SORTING THE INPUT AUTO-CORRELATION
C      TIME SERIES
C
C      INPUT PARAMETERS
C
C      IP = THE ORDER OF THE AUTO-REGRESSIVE PARAMETERS
C      IQ = THE ORDER OF THE MOVING AVERAGE PARAMETERS
C      R(I) = THE INPUT AUTO-CORRELATION VALUES
C
C      OUTPUT PARAMETERS
C      A1(I,J) = THE LEFT SIDE MODIFIED ESTIMATES
C      R1(I) = THE RIGHT SIDE MODIFIED ESTIMATES
C**********************************************************
C
       SUBROUTINE MODY(R,IP,IQ,A1,R1,M1)
       COMPLEX*16 A1(100,20),R(0:500),R1(100)
       IF(IP.EQ.M1) M2=IP
       IF(IP.LT.M1) M2=M1-IQ
       IL=IP-1
       M=IQ+1
       DO 10 I=1,M2
       DO 20 J=0,IL
       KK=IQ-J+I-1
       IF(KK.LT.0)    A1(I,J+1)=DCONJG(R(-KK))
       IF(KK.GE.0)    A1(I,J+1)=R(KK)
   20  CONTINUE
   10  CONTINUE
       DO 30 I=1,M2
       K3=M+I-1
       R1(I)=-R(K3)
   30  CONTINUE
       RETURN
       END
```

```
C************************************************************
C       THIS SUBROUTINE CONVERTS A NON-SQUARE MATRIX TO
C       A SQUARE MATRIX IN ORDER TO SOLVE ITS INVERSE
C
C       INPUT PARAMETERS
C       A1(I,J) = LEFT SIDE OF NON-SQUARE MATRIX
C       R1(I) = RIGHT SIDE OF NON-SQUARE MATRIX
C       IC = NUMBER OF COLUMNS OF THE NON-SQUARE MATRIX
C       IR = NUMBER OF ROWS OF THE NON-SQUARE MATRIX
C
C       OUTPUT PARAMETERS
C       C(I) = LEFT SIDE OF SQUARE MATRIX
C       C1(I) = RIGHT SIDE OF SQUARE MATRIX
C
C************************************************************
C
      SUBROUTINE INVR(A1,R1,IC,IR,C,C1)
      COMPLEX*16 A1(100,20),R1(100),C1(100),C(100,100)
C     COMPUTE A1 TRANSPOSE R1
      DO 120 I=1,IC
      C1(I)=(0D0,0D0)
      DO 120 J=1,IR
      C1(I)=C1(I)+DCONJG(A1(J,I))*R1(J)
  120 CONTINUE
C     COMPUTE A1 TRANSPOSE A1
      DO 130 I=1,IC
      DO 130 J=1,IC
      C(I,J)=(0D0,0D0)
      DO 130 K=1,IR
      C(I,J)=C(I,J)+DCONJG(A1(K,I))*A1(K,J)
  130 CONTINUE
      RETURN
      END
```

```
C*************************************************************
C      FILTER AR PARAMETERS AND ESTIMATE MA PARAMATERS
C
C      INPUT PARAMETERS
C      A2(I) = AR PARAMETERS
C      IP = ORDER OF AR PARAMETERS
C      X2(I) = ORIGINAL INPUT TIME SERIES
C      NPTS1 = NUMBER OF INPUT TIME SERIES
C
C      OUTPUT PARAMETERS
C      Y3(I) = FILTERED AR TIME SERIES
C      LY3 = LENGTH OF FILTERED TIME SERIES
C
C*************************************************************
       SUBROUTINE ARFIL(X2,IP,A2,Y3,LY3,NPTS1)
       REAL*8 A2(0:30),X2(0:2000),Y3(0:2000)
       M=NPTS1-IP-1
       LY3  = M+1
       A2(0)=1.0
       DO 10 NN=0,M
       Y3(NN+IP)=0D0
       DO 20 K=0,IP
       Y3(NN+IP)=Y3(NN+IP)+A2(K)*X2(NN+IP-K)
   20  CONTINUE
   10  CONTINUE
       RETURN
       END
```

```
C***************************************************
C       THIS PROGRAM CALULATES THE MA PARAMETERS
C
C       INPUT PARAMETERS
C       XF(N)= INPUT TIME SERIES
C       NPTSF= NUMBER OF POINTS
C       NQMAX= ORDER OF THE PROCESS
C       NPL = NPTSF/5 (ORDER FOR LONG AR PROCESS)
C
C       OUTPUT PARAMETERS
C       AL(I,NQMAX)= THE B(I) COEFFICIENTS
C       POW01 = THE VARIANCE
C***************************************************
C
        SUBROUTINE DURBIN(XF,NPTSF,NQMAX,NPL,AL,POW01)
        REAL*8 XF(300),AL(100,100),POW(100)
        CALL LEVIN (XF,NPTSF,NPL,AL,POW01,POW)
        XF(1)=1.0
        DO 10 I=1,NPL
   10   XF(I+1)=AL(I,NPL)
        NPL1=NPL+1
        CALL LEVIN(XF,NPL1,NQMAX,AL,POW0,POW)
        RETURN
        END
        SUBROUTINE LEVIN(X7,NPTS,NP,AYW,PYW0,PYW)
C       THIS SUBROUTINE COMPUTES THE YULE-WALKER ESTIMATES OF THE
C       AR PARAMETERS
        REAL*8 X7(300),AYW(100,100),R7(101),PYW(100)
        N2=NP+1
        DO 20 K=1,N2
        NK=NPTS-K+1
        R7(K)=0.0
        DO 10 L=1,NK
        R7(K)=R7(K)+X7(L)*X7(L+K-1)
  10    CONTINUE
        R7(K)=R7(K)/FLOAT(NPTS)
  20    CONTINUE
        AYW(1,1)=-R7(2)/R7(1)
        PYW0=R7(1)
        PYW(1)=(1.0-AYW(1,1)**2)*R7(1)
        IF(NP.EQ.1)GO TO 60
        DO 50 I=2,NP
        I1=I-1
        B=-R7(I+1)
        DO 30 K=1,I1
  30    B=B-AYW(K,I-1)*R7(I+1-K)
        AYW(I,I)=B/PYW(I-1)
        DO 40 K=1,I1
  40    AYW(K,I)=AYW(K,I-1)+AYW(I,I)*AYW(I-K,I-1)
        PYW(I)=(1.0-AYW(I,I)**2)*PYW(I-1)
  50    CONTINUE
  60    RETURN
        END
```

```
C****************************************************************
C       THIS SUBROUTINE COMPUTES THE POWER SPECTRAL DENSITY
C       OF AN ARMA PROCESS GIVEN THE MOVING AVERAGE AND
C       AUTO-REGRESSIVE COEFFICIENTS
C
C       INPUT PARAMETERS
C       IQ = ORDER OF THE MOVING AVERAGE PROCESS
C       IP = ORDER OF THE AUTO-REGRESSIVE PROCESS
C       A(I) = AUTO-REGRESSIVE COEFFICIENTS
C       B(I) = MOVING AVERAGE COEFFICIENTS
C       N = LENGTH OF THE FFT
C       VAR = THE VARIANCE OF THE MA PROCESS
C
C       OUTPUT PARAMETERS
C       PS(I) = OUTPUT POWER SPECTRUM
C
C****************************************************************
        SUBROUTINE POWER(IQ,IP,A,B,N,PS,VAR)
        REAL*8 A(0:30),B(0:30),X5(200),X6(200)
        REAL*8 Y(200),Y1(200),PS(200)
        COMPLEX*16 X8(200),X7(200)
        REAL*8 WK(1000)
        DIMENSION IWK(1001)
        ND2=N/2
        N1=N-IP-1
        DO 10 I=0,IP
        X5(I+1)=A(I)
  10    CONTINUE
        DO 15 I=1,N1
        X5(I+IP+1)=0D0
  15    CONTINUE
        CALL FFTRC(X5,N,X7,IWK,WK)
        DO 20 I=2,ND2
        X7(N+2-I)=DCONJG(X7(I))
  20    CONTINUE
        DO 25 I=1,N
        X7(I)=DCONJG(X7(I))
  25    CONTINUE
        DO 26 I=1,N
        Y(I)=CDABS(X7(I))**2
  26    CONTINUE
C       WRITE(6,60),(Y(K),K=1,N)
        N2=N-IQ-1
        DO 30 I=0,IQ
        X6(I+1)=B(I)
  30    CONTINUE
        DO 35 I=1,N2
        X6(I+IQ+1)=0D0
  35    CONTINUE
        CALL FFTRC(X6,N,X8,IWK,WK)
        DO 41 I=2,ND2
```

```
      XB(N+2-I)=DCONJG(XB(I))
41    CONTINUE
      DO 43 I=1,N
      XB(I)=DCONJG(XB(I))
43    CONTINUE
      DO 44 I=1,N
      Y1(I)=CDABS(XB(I))**2
44    CONTINUE
C     WRITE(6,60),(Y1(K),K=1,N)
      DO 50 I=1,N
      PS(I)=(Y1(I)/Y(I))*VAR
50    CONTINUE
60    FORMAT (5(3X,F9.6))
      RETURN
      END
```

DISTRIBUTION LIST (Continued)

	No. of Copies
Defense Technical Information Center Cameron Station, Alexandria, VA 22314	12
Technical Library Naval Ordnance Station, Indian Head, MD 20640	1
Library Naval Ocean Systems Center, San Diego, CA 92152	1
Technical Library Bureau of Naval Personnel, Department of the Navy Washington, DC 20370	1
Technical Library Naval Research Laboratory, Washington, DC 20375	1
Technical Library Naval Underwater Systems Center, New London Laboratory, New London, CT 06320	1
Technical Library Naval Underwater Systems Center, Newport Laboratory Newport, RI 02840	1
Technical Library Naval Coastal Systems Center, Panama City, FL 32401	1
Applied Mathematics Laboratory David Taylor Naval Ship Research and Development Center, Attn: Librarian, Bethesda, MD 20084	1
Mr. James Smith Code 340R, Naval Air Systems Command, Washington, DC 20361	1
Dr. Thomas Ballard Code U22, Naval Surface Weapons Center, White Oak, MD 20950	1
Dr. S. S. Rao Department of Electrical Engineering, Villanova University, Villanova, PA 19085	1
Dr. Donald W. Tufts Department of Electrical Engineering, University of Rhode Island, Kingston, RI 02881	1

DISTRIBUTION LIST (Continued)

	No. of Copies
Dr. Steven M. Kay Department of Electrical Engineering, University of Rhode Island, Kingston, RI 02881	1
Mr. Donato Russo Code 303, Naval Air Development Center, Warminster, PA 18974	1
Mr. Robert E. Imbrogno Code 2042, Naval Air Development Center, Warminster, PA 18974	1
Mr. Nicholas Tavani Code 3034, Naval Air Development, Warminster, PA 18974	1
Technical Library Naval Surface Weapons Center, White Oak, MD 20950	1
NAVAIRDEVCEN (Code 8131)	3
NAVAIRDEVCEN (Code 3041)	6

DISTRIBUTION LIST

REPORT NO. NADC-84027-30

	No. of Copies
Commander, Naval Air Systems Command (AIR-00D4) Washington, DC (2 for retention) (2 forAIR-340, Mr. T. Guarini, Mr. W. Parigian) (2 for AIR-5490, CDR. G. Plum, Mr. P. Koenig)	6
Chief of Naval Operations (OP-951C)	1
Project Manager, Anti-Submarine Warfare Systems Washington, DC 20362 (Mr. L. Fabian, Mr. W. Richter)	2
Navy Library, National Space Technology Laboratory Attn: Navy Librarian, Bay St. Louis, MS 39522	1
U.S. Army Research Office P.O. Box 12221, Attn: Librarian, Research Triangle Park, NC 27706	1
Director, National Security Agency National Security Agency, Attn: Librarian, Fort Meade, MD 20755	1
ATAA-SL, Library, U. S. Army TRADOC Systems Analysis Activity Department of the Army, White Sands Missile Range, NM 88022	
ARI Field Unit — USAREUR Attn: Library, c/o ODCSPER HQ USAEREUR & 7th Army, APO New York 09403	1
Library, Code 1424 Naval Postgraduate School, Monterey, CA 93940	1
Technical Information Division Naval Research Laboratory, Washington, DC 20375	1
Nimitz Library Attn: Librarian, U.S. Naval Academy, Annapolis, MD 21402	1
Statistics and Probability Program (Code 411 (SP)), Office of Naval Research, Arlington, VA 22217	1

B-5

END

DATE
FILMED

0-84

DTIC